

# **INTERACTIVE VIDEO RETRIEVAL USING IMPLICIT USER FEEDBACK**

**Stefanos Vrochidis**

Doctor of Philosophy (PhD) Thesis

Supervisors:

Dr Ioannis Patras

Dr Ioannis Kompatsiaris

*For my wife Giota and my son Giorgos,  
my beloved Dad and Mum, and my grandparents Stefanos and Ourania*

## **ACKNOWLEDGMENTS**

Although the time I have decided to start this challenging journey, I have been already working as a researcher, I have to admit that writing this thesis was one of the most inspiring and interesting experiences of my life. Since this journey is coming to an end, I would like to pay a tribute to the people, who constantly supported me during this period and without their help this goal would have never been fulfilled.

First of all, I would like to thank my supervisors Ioannis Patras and Ioannis Kompatsaris for their helpful discussions, scientific guidance and constructive feedback throughout the past four years. I would like also to thank them for their encouragement, support and overall for being great supervisors and friends.

I would also like to thank my examiners, Joemon Jose and Apostolos Georgakis for their constructive feedback and comments during my viva examination.

In addition, I would like to thank the past and current members of Multimedia Group of the Information Technologies Institute for their kind support and interest in my research and for providing a friendly working environment and fruitful collaboration.

I would also like to thank the members of Multimedia and Vision Research Group of Queen Mary, University of London for facilitating my stay at London and for participating so enthusiastically in my user experiments. Special thanks should go to Navid Hajimirza for his guidance and help for setting up the eye-tracking experiment.

I also owe many thanks to my friends who accompanied me during these years, supporting my efforts towards the completion of this journey and for making my life more enjoyable.

Finally, I would like to thank my wife for her constant encouragement and motivation, as well as my mother and my parents in law for their unconditional support and patience during this period.

## **ABSTRACT**

In the recent years, the rapid development of digital technologies and the low cost of recording media have led to a great increase in the availability of multimedia content worldwide. This availability places the demand for the development of advanced search engines. Traditionally, manual annotation of video was one of the usual practices to support retrieval. However, the vast amounts of multimedia content make such practices very expensive in terms of human effort. At the same time, the availability of low cost wearable sensors delivers a plethora of user-machine interaction data. Therefore, there is an important challenge of exploiting implicit user feedback (such as user navigation patterns and eye movements) during interactive multimedia retrieval sessions with a view to improving video search engines. In this thesis, we focus on automatically annotating video content by exploiting aggregated implicit feedback of past users expressed as click-through data and gaze movements. Towards this goal, we have conducted interactive video retrieval experiments, in order to collect click-through and eye movement data in not strictly controlled environments. First, we generate semantic relations between the multimedia items by proposing a graph representation of aggregated past interaction data and exploit them to generate recommendations, as well as to improve content-based search. Then, we investigate the role of user gaze movements in interactive video retrieval and propose a methodology for inferring user interest by employing support vector machines and gaze movement-based features. Finally, we propose an automatic video annotation framework, which combines query clustering into topics by constructing gaze movement-driven random forests and temporally enhanced dominant sets, as well as video shot classification for predicting the relevance of viewed items with respect to a topic. The results show that exploiting heterogeneous implicit feedback from past users is of added value for future users of interactive video retrieval systems.



**TABLE OF CONTENTS**

Chapter 1	Introduction .....	1
1.1.	Motivation .....	1
1.2.	Research objectives and approach .....	4
1.3.	Research contribution .....	7
1.4.	Thesis outline .....	8
1.5.	Publications .....	8
1.5.1.	Included publications .....	8
1.5.2.	Related publications .....	9
Chapter 2	Literature review .....	11
2.1.	Video indexing and retrieval .....	11
2.1.1.	Video indexing .....	13
2.1.1.1.	Video shot segmentation .....	13
2.1.1.2.	Keyframe extraction .....	14
2.1.1.3.	Low-level features extraction .....	15
2.1.1.4.	High-level concepts and events .....	16
2.1.1.5.	Textual metadata extraction .....	17
2.1.2.	Video retrieval .....	18
2.1.2.1.	Retrieval functionalities .....	18
2.1.2.2.	Relevance feedback .....	20
2.1.2.3.	Retrieval interfaces .....	20
2.1.2.4.	Video retrieval evaluation .....	22
2.2.	Implicit user feedback in information retrieval .....	25
2.2.1.	Exploitation of past user interaction in information retrieval .....	26
2.2.1.1.	Exploitation of past user interaction in textual retrieval .....	27
2.2.1.2.	Past user interaction-based approaches in multimedia retrieval .....	28
2.2.2.	Eye movement-based approaches .....	30
2.2.2.1.	Eye movement-based works in textual retrieval .....	30
2.2.2.2.	Eye movement-based approaches in multimedia retrieval .....	32
2.2.3.	Additional implicit feedback approaches in multimedia retrieval .....	36

2.2.3.1.	Cognitive implicit tagging .....	36
2.2.3.2.	Affective-based tagging.....	38
<b>Chapter 3</b>	<b>Video content analysis .....</b>	<b>40</b>
3.1.	Introduction .....	40
3.2.	Dataset.....	41
3.3.	Video indexing .....	42
3.3.1.	Temporal indexing .....	42
3.3.1.1.	Shot segmentation .....	43
3.3.1.2.	Keyframe extraction .....	46
3.3.2.	Textual indexing .....	47
3.3.3.	Visual similarity indexing .....	48
3.3.3.1.	Visual descriptor extraction .....	49
3.3.3.2.	Indexing of multidimensional vectors .....	50
3.3.3.3.	Ranking and retrieval .....	52
3.4.	LELANTUS interactive video search engine.....	53
3.4.1.	Interface.....	53
3.4.2.	Implementation insights .....	55
3.5.	Conclusions .....	56
<b>Chapter 4</b>	<b>Exploitation of user past navigation patterns to enhance interactive video retrieval .....</b>	<b>57</b>
4.1.	Introduction .....	57
4.2.	Implicit feedback analysis.....	60
4.2.1.	Implicit interest indicators .....	60
4.2.2.	Action graph .....	61
4.2.3.	Weighted graphs .....	65
4.2.4.	Generation of recommendations .....	66
4.3.	Combining visual and implicit feedback information .....	68
4.3.1.	Training set construction .....	69
4.3.2.	Support vector machine classifier .....	71
4.4.	Experiments and results .....	73
4.4.1.	Interactive video retrieval framework .....	73

4.4.2.	Training phase .....	74
4.4.3.	Recommendations evaluation .....	75
4.4.4.	Visual search optimisation experiment .....	78
4.5.	Interaction modes .....	82
4.6.	Conclusions .....	86
Chapter 5 Investigating aggregated gaze-based implicit feedback in interactive video retrieval .....		87
5.1.	Introduction .....	87
5.2.	Gaze-movements analysis .....	89
5.2.1.	Eye movements .....	89
5.2.2.	Fixation analysis in video retrieval .....	91
5.2.3.	Pupil dilation analysis .....	93
5.2.4.	Feature extraction.....	94
5.2.5.	Classification using support vector machines .....	95
5.3.	Experiments.....	97
5.3.1.	Experimental setup.....	97
5.3.2.	Training phase .....	100
5.3.3.	Testing phase .....	101
5.4.	Results and evaluation .....	101
5.4.1.	Quantitative evaluation.....	101
5.4.1.1.	Feature performance.....	102
5.4.1.2.	Evaluating gaze data aggregation .....	105
5.4.2.	Visual assessment of results .....	110
5.5.	Conclusions .....	111
Chapter 6 Automatic video annotation based on query clustering and eye movements.....		113
6.1.	Introduction .....	113
6.2.	Video annotation framework.....	116
6.3.	Dominant set query clustering using temporal information.....	117
6.3.1.	Dominant set clustering.....	118
6.3.2.	Query similarity .....	119
6.3.2.1.	WordNet-based similarity.....	120

6.3.2.2.	Temporally enhanced similarity .....	121
6.3.2.3.	Smoothing process .....	121
6.4.	Query clustering based on gaze-driven random forests .....	122
6.4.1.	Random forests .....	124
6.4.1.1.	Construction of random forests .....	124
6.4.1.2.	Impurity function .....	125
6.4.1.3.	Predicting classes with random forests .....	126
6.4.1.4.	Advantages and disadvantages of random forests .....	127
6.4.2.	Unsupervised random forests .....	127
6.4.2.1.	Random forest dissimilarity .....	128
6.4.3.	Gaze-driven random forests .....	129
6.4.3.1.	Affinity matrix .....	131
6.4.3.2.	Splitting criterion for decision tree construction .....	136
6.5.	Results and evaluation .....	138
6.5.1.	Clustering evaluation methodology .....	139
6.5.2.	Training and testing .....	141
6.5.1.	Results of topic-based merging .....	142
6.5.2.	Results of dominant set clustering .....	143
6.5.2.1.	Evaluation of clustering .....	143
6.5.2.2.	Classification evaluation .....	143
6.5.2.3.	Annotated shots .....	144
6.5.3.	Results and evaluation for gaze-driven random forest clustering .....	145
6.5.3.1.	Classification performance .....	151
6.5.3.2.	Annotated shots .....	151
6.5.4.	Comparison of annotations .....	151
6.6.	Conclusions .....	153
Chapter 7	Conclusions .....	155
7.1.	Summary of achievements .....	155
7.2.	Future work .....	157
7.2.1.	On line video retrieval system based on implicit user feedback .....	158
7.2.2.	Detect user interest based on implicit feedback .....	159

7.2.3. Dynamic multimedia content modelling .....	160
Bibliography .....	161

**LIST OF FIGURES**

Figure 1.1. Eye tracker.....	3
Figure 1.2. Conceptual framework.....	5
Figure 2.1. Video indexing framework.....	13
Figure 2.2. VERGE video search engine .....	21
Figure 2.3. 2x2 latin square design, in which two users are searching for two topics with two system variants .....	25
Figure 3.1. Video indexing and retrieval framework.....	41
Figure 3.2. Example keyframes of the TRECVID 2008 test video set.....	42
Figure 3.3. Shot segmentation framework .....	43
Figure 3.4. Example of shot and associated ASR.....	47
Figure 3.5. Search engine interface .....	52
Figure 3.6. Keyframe-based video representation.....	54
Figure 3.7. LELANTUS interface showing temporally adjacent shots.....	55
Figure 4.1. Search session and subsessions.....	62
Figure 4.2. Classification of user actions .....	63
Figure 4.3. Action graph after user interaction.....	64
Figure 4.4. Weighted graph after processing the action graph of Figure 4.3.....	65
Figure 4.5. Algorithm to combine visual features and implicit user feedback ...	69
Figure 4.6. Video indexing and retrieval framework.....	73
Figure 4.7. Precision for the results of the baseline and enhanced systems.....	78
Figure 4.8. Recall for the results of the baseline and enhanced systems .....	78
Figure 4.9. A visual representation of the user preference on the hybrid and visual rankings .....	80
Figure 4.10. A visual representation of the clicks on the hybrid and visual ranking .....	80

Figure 4.11. Histogram of the clicks. The horizontal axis stands for the number of clicks (positive for hybrid ranking and negative for visual), while the vertical for the frequency.....	81
Figure 4.12. The user submits a textual query with the keyword “water” searching in the ASR transcripts.....	82
Figure 4.13. Results from a textual query (keyword “water”) searching with the aid of the weighted graph .....	83
Figure 4.14. Query by image example. Content-based analysis is employed for this query. The input image is the one on the left top corner .....	83
Figure 4.15. Query by image example. Relations from the weighted graph are used to realise the query. The input image is the one on the left top corner. .	84
Figure 4.16. Hybrid search combining visual features and implicit user feedback. The input image is the one on the left top corner.....	84
Figure 4.17. Combined Ranking of results for the query shot on the top left corner.....	85
Figure 4.18. Keyword suggestions in a text query (left) and in an image by example query (right) .....	86
Figure 5.1. The user is searching for video scenes depicting books. The fixations are presented as blue spots on the interface.....	92
Figure 5.2. Pupil dilation of a user during a fixation.....	94
Figure 5.3. A schematic view of the experiment.....	98
Figure 5.4. FaceLAB eye-tracker .....	99
Figure 5.5. Accuracy of the classifier for topics A-D for the 3 feature sets .....	104
Figure 5.6. Accuracy of the classifier for topics A-D for the 3 feature sets .....	105
Figure 5.7. Precision-Recall curve for model 1 when the ratio $w_p/w_n$ (points on the P-R curve) takes values from 0.2 to 10.....	107
Figure 5.8. The precision and recall for model 2 are presented, when 1, 2 and 3 users are considered.....	108

Figure 5.9. The F-Score for model 2 are presented .....	109
Figure 5.10. The EER for model 2 is reported, when we use aggregated data from 1,2 and 3 test users respectively.....	109
Figure 5.11. Shots of interest for topic A (Find shots of one or more people with one or more horses).....	110
Figure 5.12. Shots of interest for topic B (Find shots of a map) .....	111
Figure 5.13. Shots of interest for topic C (Find shots of one or more people with one or more books).....	112
Figure 5.14. Shots of interest for topic D (Find shots of food and/or drinks on a table).....	112
Figure 6.1. Video annotation framework .....	116
Figure 6.2. Search session and queries .....	118
Figure 6.3. Search sessions by several users in the temporal and semantic similarity dimension. The large clusters indicate semantic topics, while the smaller ones search sessions deal with these topics.....	122
Figure 6.4. Random forest generation.....	125
Figure 6.5. Decision tree construction in gaze-driven random forests .....	130
Figure 6.6. Comparison of subsessions. On the top the direct comparison between the textual queries is illustrated. The common images identified in the dependent queries are shown in green circles. ....	132
Figure 6.7. Sets A and B in a bipartite graph representation. Distances for non duplicate shots are represented with red dashed edges, while black solid edges indicate distances between near duplicates.....	133
Figure 6.8. Interactive experiment and video annotation.....	139
Figure 6.9. Automatic annotations using model 7 .....	144
Figure 6.10. The gaze-driven RF NMI performance for 4 clusters.....	147
Figure 6.11. The gaze-driven RF NMI performance for 6 clusters.....	147
Figure 6.12. The gaze-driven RF NMI performance for 8 clusters.....	148



Figure 6.13. The gaze-driven RF NMI performance for 10 clusters .....	149
Figure 6.14. The performance for the 3 clustering algorithms.....	150
Figure 6.15. The average performance for the 3 clustering algorithms.....	150
Figure 6.16. F-Score performance for different merging approaches .....	152
Figure 6.17. Classification and final precision for different clustering methods. .....	153

**LIST OF TABLES**

Table 4.1. Assign weights for each action .....	61
Table 4.2. User-topic assignments.....	74
Table 4.3. Numerical statistics for the weighted graph .....	75
Table 4.4. Latin square user experiment .....	76
Table 4.5. Topic search sequences .....	77
Table 4.6. Precision and recall for the baseline and enhanced systems.....	77
Table 4.7. Pairwise comparison of the hybrid retrieval function with the visual one .....	79
Table 4.8. Clicks on hybrid and visual ranking.....	80
Table 5.1. Eye movement-based features .....	96
Table 5.2. Training cases .....	100
Table 5.3. First case (features 1-5) .....	102
Table 5.4. Second case (features 1-5).....	103
Table 5.5. Third case (features 1-7).....	103
Table 5.6. Forth case (features 1-9).....	103
Table 5.7. Performance of the classifier for each user.....	106
Table 5.8. Average results per topic for training case 1 .....	106
Table 5.9. Average results per user for training case 1.....	107
Table 5.10. Model 2 when data of one, two and three users are aggregated .....	108
Table 5.11. P@18 for the topics A-D .....	110
Table 6.1. Dominant set clustering algorithm .....	119
Table 6.2. Visually enhanced Jaccard similarity algorithm .....	135
Table 6.3. Training and testing cases.....	141
Table 6.4. First case (Topic-based merging).....	142

Table 6.5. Second case (Cluster-based merging) .....	143
Table 6.6. Second case (Cluster-based merging) .....	144
Table 6.7. Produced annotations for second cases and training data from users 1-5 and 4-8 .....	145
Table 6.8. NMI for the gaze-driven RF for 4 clusters .....	146
Table 6.9. NMI for the gaze-driven RF for 6 clusters .....	147
Table 6.10. NMI for the gaze-driven RF for 8 clusters .....	148
Table 6.11. NMI for the gaze-driven RF for 10 clusters.....	148
Table 6.12. NMI comparison for the 3 clustering techniques for 500 trees.....	149
Table 6.13. Classification performance of the forth case (gaze-driven RF cluster-based merging).....	150
Table 6.14. Annotations of the forth case (gaze-driven RF cluster-based merging) .....	151
Table 6.15. Results for different training user data variations .....	152
Table 6.16. Average produced annotations for second and forth training cases and training data from users 1-5 and 4-8.....	152

## **Chapter 1**

### **INTRODUCTION**

*This chapter summarises the recent challenges in interactive video retrieval and discusses the motivation that triggered our research activities. Then, the objectives of this research and the proposed approach are presented. Finally, we summarise the contribution of this work and report the achieved publications.*

#### **1.1. Motivation**

In the recent years, the rapid development of digital technologies, the low cost of recording media, as well as the growth of communication networks have led to a rapid increase in the availability of multimedia content worldwide. The availability of such content, as well as the increasing user need of searching into multimedia collections place the demand for the development of advanced multimedia search engines that integrate multimodal retrieval techniques. Therefore, video retrieval remains one of the most challenging tasks of research. Despite the significant advances in this area recently, further advancements in multiple fields of video retrieval are required to improve the performance of video search engines. More specifically, major research challenges are still notable in the areas of semantic search with concept detection, multi-modal analysis and retrieval algorithms, as well as interactive search and relevance feedback (Lew, et al. 2006).

In parallel, the high Internet penetration and the increasing usage of social platforms and environments for content exchange have generated tremendous amounts of user-machine interaction data, while the easily accessible and affordable technologies for recording context-based information, as well as the increasing usage of biometric and human behaviour recording devices, have

given easy access to human behavioural data. This fact reveals the need for the development of enhanced search techniques that could exploit the aforementioned information and combine it with content-based modalities, in order to generate additional metadata and facilitate the access to multimedia content.

Video as a medium includes rich heterogeneous information, such as sound, text, as well as sequences of still images. Hence, current approaches of video retrieval adapt and combine techniques from text and image retrieval fields and employ multimodal approaches to deal with such diverse and heterogeneous information. To perform video retrieval, it is essential to index the content by creating efficient representations and descriptions of the video source. Shot change detection is the initial step of video segmentation and indexing (Lew, et al. 2006), in order to split the initial video to smaller scenes (i.e. video shots). By processing the audiovisual data, it is possible to extract low-level features (Sebe, et al. 2003) for each shot, however due to the well known problem of the semantic gap (i.e. the difficulty in translating low-level features to human understandable concepts), it is difficult to convert them to meaningful high-level concepts. Combination and fusion of heterogeneous information (i.e. visual and textual) is a first step towards the solution of this problem, as promising results have been presented both in image and video retrieval (Kherfi, D. and D. 2004), (Chang, Manmatha and Chua 2005), (Vrochidis, et al. 2008), (Snoek and Worring 2005), however the semantic representation and indexing of the multimedia content has not yet managed to overcome the semantic gap.

An alternative way to bridge the semantic gap is to take advantage of the implicit and explicit feedback provided by the users (Hopfgartner, et al. 2008) of an interactive video search engine. During interactive video retrieval tasks, multiple sessions take place, in which the user interacts with the system either directly by submitting queries, browsing the video source and providing explicit relevance feedback by selecting specific shots of interest, or indirectly through his/her involuntary reactions (e.g. eye movements, facial expressions, etc.). Relevance feedback-based techniques in information retrieval (IR) constitute complementary methods to further improve the performance of a system by

requesting usually explicit feedback from the user (i.e. positive and negative examples). Despite promising results in image and video retrieval (Zhou, et al. 2003), (Giacinto and Roli 2005), (Gurrin, et al. 2006), explicit relevance feedback-based functionalities are not very user-popular due to the fact that users are usually reluctant to provide such a feedback. Motivated by this, we propose to take into account the implicit feedback provided to the system by the users during the search process.



**Figure 1.1. Eye tracker**

As implicit user feedback we consider any voluntary or involuntary behaviour of the user during the interactive query session. In a typical search engine, when the user searches for a video, she/he navigates through the content and submits different queries by performing different mouse movements and clicks, as well as keystrokes that can provide information of his/her preferences. In addition, the physical involuntary reactions of the user as eye movements, heart rates, brain neuron reactions could also be considered as implicit feedback. Recording of the physical reactions of the user usually requires specialised equipment such as eye trackers (Figure 1.1), which are devices capturing the user eye movements, as well as biometric and electroencephalography sensors. The main advantage of using implicit techniques is that they do not require the user to provide explicit feedback. Although implicit information is in general thought to be less accurate than explicit (Nichols 1998) it has the advantage that large quantities of past user interaction data (e.g. log files in web search engines) can be gathered at no extra effort to the user.

Exploiting implicit user feedback in multimedia retrieval would have an important impact in a variety of applications. First, it could facilitate video search

and retrieval in web applications, which are widely used by everyday users. For instance New York Times reported that YouTube<sup>1</sup> has reached 2 billion searches per day in 2010. Therefore, the exploitation of such large amounts of user interaction data could result to a significant improvement of the search engine experience. Furthermore, given the fact that personal digital collections (e.g. photos and videos) have grown exponentially in the recent years, implicit feedback techniques could also contribute to generating additional semantic relations between the content, supporting that way the information retrieval tasks. Social media platforms, which already integrate several multimedia retrieval functionalities, would also benefit from such technologies, since the user interaction data during content browsing and exchange could be processed to facilitate search tasks. Finally, implicit user feedback approaches could be considered of added value in the domains of professional search (e.g. patent search), in which the search session data of past users could be reused in an unobtrusively manner to support future retrieval tasks.

Overall, it can be said that the effective application of implicit feedback techniques could be a way to overcome current limitations of multimedia annotation and retrieval approaches that rely upon explicit user feedback. However, the distillation of meaningful information from noisy user interaction data can be considered as an important challenge to be addressed.

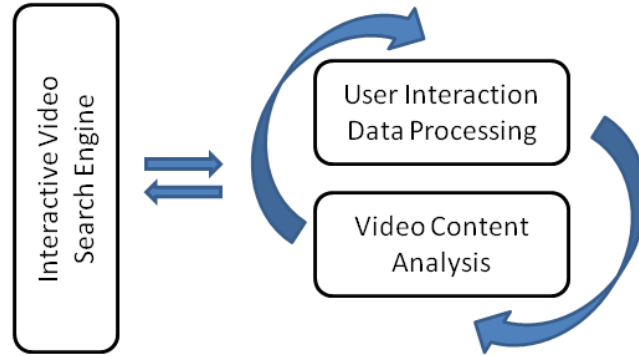
## **1.2. Research objectives and approach**

In this work we propose to exploit the implicit user feedback inferred by the user interaction patterns and the gaze movements. The research objective of this thesis is to exploit past aggregated user implicit feedback, in order to generate additional metadata for a given dataset, as well as to build predictor models that would be capable of judging the relevance of a shot to a query based on

---

<sup>1</sup> <http://www.youtube.com/>

aggregated user interaction. To this end we combine several techniques including video content analysis, text processing, supervised and unsupervised machine learning (i.e. classification and clustering), as well as heterogeneous representation schemes such as graphs and vectors. The conceptual framework of this approach is illustrated in Figure 1.2, in which the user interaction data are combined with video content analysis to facilitate interactive video retrieval.



**Figure 1.2. Conceptual framework**

In the context of establishing the environment for this research, we first perform video analysis and processing by employing well established techniques from image and video analysis. Then, we implement an interactive video retrieval engine, which is used to conduct user experiments.

With a view to exploiting user navigation patterns, we generate additional semantic relations between the multimedia items (e.g. shots) included in a video collection and optimise content-based retrieval. This is performed by combining the information extracted from video content with aggregated past user navigation patterns. In this context, we take into account the user navigation pattern during a video retrieval task, which is expressed by user actions such as mouse clicks and keyboard inputs. These data are used to construct an action graph that describes the navigation of the user during the search process, by employing a methodology that defines several search subsessions (i.e. parts of sessions that the user is considered to search for the same topic). Subsequently, this graph is converted to a weighted graph that initiates relations between the queries and the content and is used to perform retrieval. Furthermore, this graph



is exploited, during a query by visual example to define positive and negative samples, in order to drive a pseudo-relevance feedback modality based on Support Vector Machines (SVM). The evaluation of the system is performed by conducting real user experiments with an interactive video search engine.

The next step in our research focuses on the investigation of the role of aggregated gaze movements in interactive video retrieval. We propose an approach, in which, the gaze movements of past users are processed, in order to extract fixations (i.e. the eye remains fixed on a specific point for a certain amount of time) and pupil dilations. Then, we extract a set of features that describes each video shot based on aggregated fixation and pupil dilation characteristics. Subsequently, we employ SVMs and we train models that could predict which fixations of a future user could be considered as indicators of interest. We evaluate this approach by conducting an experiment, in which users are recruited to perform video retrieval with an interactive search engine, while their gaze movements are captured with an eye tracker.

Finally, we propose an automatic annotation framework for video content by exploiting implicit user feedback during interactive video retrieval, as this is expressed with gaze movements, mouse clicks and queries. The queries submitted by new users are considered unknown and they are grouped in search topics and using two different clustering methods. First, we employ a dominant set clustering approach, in which we take into account the semantic similarity between the submitted queries and the temporal dimension to create query clusters. Then, we present a technique based on random forests clustering, in which the construction of the decision trees is driven by the user gaze movements, as well as by the semantic similarity between the queries. The evaluation shows that the combination of aggregated click and gaze movement data can be utilised effectively for automatic video tagging and annotation purposes.

### 1.3. Research contribution

The research contribution of the work is split into three parts and is presented in Chapters 4-6.

First, a notable contribution of this work is the methodology of graph analysis based on subsessions, as well as the approach for combining visual features with implicit user feedback under a supervised machine learning framework, which adds a semantic flavour to visual search. This approach enhances existing works (e.g. (Hopfgartner, et al. 2008), (Yang, et al. 2007)), in the area by constructing several sub-graphs based on subsession definition, instead of a single graph and by combining graph-structured past user interaction with content-based modalities.

An additional research contribution of this thesis is the investigation of the role of aggregated gaze movements in not strictly controlled environments by conducting interactive video retrieval experiments. In this context, we propose a novel methodology for identifying shots of interest for new users, who search for a new query based on aggregated gaze data of past users combining a variety of eye movement features. This work goes beyond the state of the art showing that the combination of aggregated fixation and pupil dilation-based features from past users could effectively be applied to detect user interest for new users in not strictly controlled environments. Existing approaches (e.g. (Klami, et al. 2008)), have been mostly relying upon fixation-based features and consider more controlled environments compared to our approach.

Finally, an important contribution of the proposed thesis is the video annotation framework, which is based on unsupervised (clustering) and supervised (classification) machine learning. The first approach is based on temporally enhanced dominant set clustering, while the second includes a novel variation of random forests algorithm, which integrates the gaze movements in the decision tree construction. This approach provides an alternative solution to the problem of query classification, where the semantic categories are not predefined, by enhancing the query clustering process with temporal and gaze movement data, while existing works in the area (e.g. (Beitzel, et al. 2007), (Wen, et al. 2002))

deal either with predefined query categories or they consider only click-through data. Furthermore, we integrate this novel clustering technique and the aforementioned gaze-based interest detection approach in an automatic video annotation framework.

#### **1.4. Thesis outline**

This thesis is structured as follows:

- Chapter 2 deals with the literature review and the state of the art both in the areas of interactive video retrieval and in user implicit feedback exploitation during information retrieval (IR) tasks.
- Chapter 3 presents the video analysis we perform in the context of our research and the implementation of an interactive video search engine.
- Chapter 4 discusses the methodology for processing patterns of user interaction and combine them with content-based modalities.
- Chapter 5 presents the eye-tracking experiment and the methodology for analysing aggregated gaze data.
- Chapter 6 proposes a framework for video annotation that combines queries clustering and gaze movement-based analysis.
- Chapter 7 concludes this thesis and proposes future work.

#### **1.5. Publications**

Several parts of the work and the results presented in this thesis are included in various research publications. Furthermore, additional papers have been published that are related and lead to the proposed thesis. These papers are listed below:

##### **1.5.1. Included publications**

Vrochidis, S., Patras, I., Kompatsiaris, I., "Exploiting gaze movements for automatic video annotation" *Proceedings of the 13<sup>th</sup> International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2012)*, Dublin, Ireland, 2012.

Vrochidis, S., Kompatsiaris, I., Patras, I., "Utilizing Implicit User Feedback to Improve Interactive Video Retrieval". *Advances in Multimedia*, Hindawi, vol. 2011, Article ID 310762, 18 pages, 2011. doi:10.1155/2011/310762.

Vrochidis, S., Patras I., Kompatsiaris, I. "An Eye-tracking-based Approach to Facilitate Interactive Video Search", *Proceedings of 2011 ACM International Conference on Multimedia Retrieval (ICMR2011)*, Trento, Italy, 2011.

Vrochidis, S., Kompatsiaris, I., Patras, I., "Optimizing Visual Search with Implicit User Feedback in Interactive Video Retrieval", *Proceedings of the ACM International Conference on Image and Video Retrieval (CIVR 2010)*, pp 274-281, Xi'an, China, 2010.

Vrochidis, S., Kompatsiaris, I., Patras, I., "Exploiting Implicit User Feedback in Interactive Video Retrieval", *Proceedings of the 11th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2010)*, Desenzano del Garda, Italy, 2010.

### **1.5.2. Related publications**

Moumtzidou, A., Sidiropoulos, P., Vrochidis, S., Gkalelis, N., Nikolopoulos, S., Mezaris, V., Kompatsiaris, I., Patras, I., "ITI-CERTH participation to TRECVID 2011", *Proceedings of TRECVID 2011 Workshop*, Gaithersburg, MD, USA, 2011.

Moumtzidou, A., Dimou, Gkalelis, N., Vrochidis, S., Mezaris, V., Kompatsiaris, I., "ITI-CERTH participation to TRECVID 2010", *Proceedings of TRECVID 2010 Workshop*, Gaithersburg, MD, USA, 2010.

Vrochidis, S., Moumtzidou, A., King, P., Dimou, A., Mezaris V., Kompatsiaris, I., "VERGE: A video interactive retrieval engine", *Proceedings of the 8th International Workshop on Content-Based Multimedia Indexing (CBMI 2010)*, pp. 142-147, Grenoble, France, 2010.

Moumtzidou, A., Dimou, A., King, P., Vrochidis, S., Angeletou, A., Mezaris, V., Nikolopoulos, S., Kompatsiaris, I., Makris, L., "ITI-CERTH participation to TRECVID 2009 HLF and Search", *Proceedings of TRECVID 2009 Workshop*, Gaithersburg, MD, USA, 2009.

Vrochidis, S., King, P., Makris, L., Moutzidou, A., Nikolopoulos, S., Dimou, A., Mezaris, V., Kompatsiaris, I., "MKLab Interactive Video Retrieval System", *Proceedings of ACM International Conference on Image and Video Retrieval (CIVR09)* - VideOlympics Showcase event, Santorini, Greece, 2009.

## **Chapter 2**

### **LITERATURE REVIEW**

*Interactive video retrieval based on implicit user feedback mostly considers two important dimensions: video indexing and retrieval, as well as the user feedback and interaction with the search engine. First, this chapter discusses the related work in the area of interactive video indexing and retrieval from the content-based perspective. Specifically, we present the main techniques employed for video indexing including shot segmentation and feature extraction, while we discuss video retrieval by presenting the most common search functionalities, the retrieval interfaces and the evaluation methodologies. Then, the chapter focuses on the exploitation of implicit user feedback during information retrieval tasks. In this context, we first present the works that consider user navigation patterns during search both in the textual and multimedia domains. Then we discuss the implicit feedback approaches that deal with eye movements and finally we briefly present other relevant works in the area focusing on affective retrieval and implicit tagging using other forms of user implicit feedback such as brain neuron reactions.*

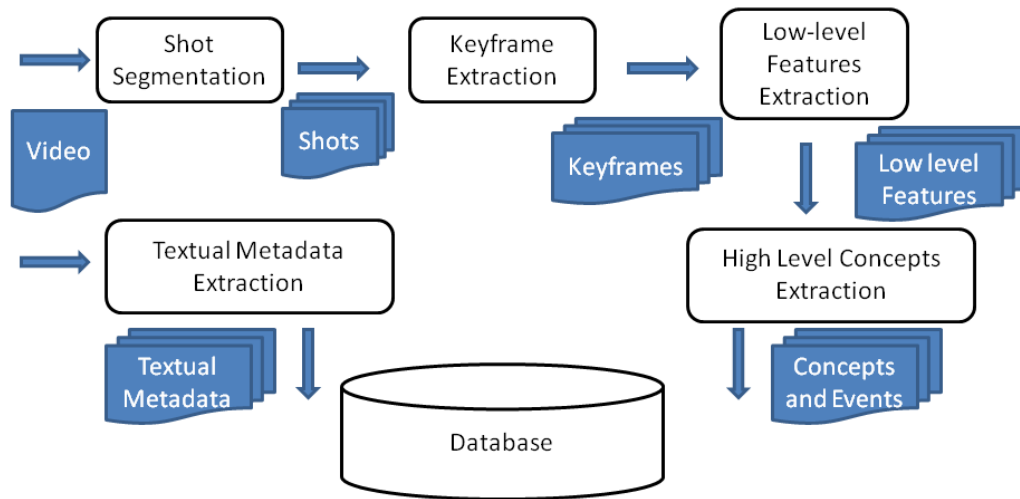
#### **2.1. Video indexing and retrieval**

Interactive video retrieval research deals with facilitating user access to video collections through the development of more advanced retrieval techniques and systems. The objective of interactive video retrieval research is to improve the retrieval experience for users interacting with video content in terms of system effectiveness and efficiency for the tasks they wish to accomplish, as well as in terms of user satisfaction.

Video indexing and retrieval is typically performed at shot level (Over et al. 2009). As shot we consider a video sequence, which is a set of consecutive series of frames that constitutes a unit of action in a film. Practically, it is a part of the video that has been taken without interruption by a single camera. Therefore, there is a need for shot segmentation, which is dictated both by the significant variability in the video content (e.g. movies, documentaries, sports), in order to separately index each elementary temporal segment of it. Based on this shot-oriented analysis, interactive video retrieval provides the user with a set of functionalities for facilitating searching, browsing and navigating within large collections of video shots.

To enable content-based retrieval of video shots, a variety of metadata generation techniques are employed. First, motion features are extracted from each shot based on the identification of salient points and their trajectories in time. In addition, image processing techniques are applied after extracting representative keyframes from the video shots and subsequently low-level visual features are generated. On top of these, in the recent years, the research trend has moved towards the extraction of high-level and human understandable concepts and events. Specifically, the concept and event-based indexing of video as a research direction has started to receive particular attention, following studies in neuroscience, which showed that humans perceive real life using past experience structure in events (Zacks, et al. 2001). Finally, textual metadata are generated by applying automatic speech recognition on the audio part of the video, by processing captions, as well as by applying Optical Character Recognition (OCR) techniques to identify text on the keyframes.

Based on the aforementioned techniques and/or the combination of them, several video retrieval systems have been developed. In the following, we will present the state of the art in this area by discussing the video shot-based indexing methods, as well as video retrieval techniques, functionalities and interfaces. Since the area of video indexing and retrieval is very broad, we mostly focus on the parts that are more relevant to the proposed thesis, and we present the most recent and well performing algorithms that fall in this area.



**Figure 2.1. Video indexing framework**

### 2.1.1. Video indexing

Video indexing consists of several image and video analysis techniques. Figure 2.1 illustrates a typical video indexing framework. The video source is first segmented into shots and subsequently the most representative keyframe is extracted. In the following, low-level visual and/or motion features are generated from the shots and the keyframes. This information is further exploited using machine learning techniques to extract high-level concept and events. In parallel, textual metadata are extracted by processing the audio part of the video, the close captions and by applying OCR on the keyframes.

#### 2.1.1.1. Video shot segmentation

Video shot segmentation is based on shot boundary detection. The basic idea of shot boundary detection to perform segmentation is frame similarity. It is obvious that consecutive frames, which belong to the same shot, are visually similar. On the other hand, consecutive frames that are assigned to different and temporally neighbouring shots are quite different from the visual perspective. Of course there are cases, in which consecutive frame similarity does not provide enough information regarding the shot cut existence. For instance, when the camera moves very fast, the similarity of consecutive frames decreases significantly. In addition, there are cases, in which shot cuts are not easily



distinguishable. This occurs in the case of special transitions such as fade in/out, dissolve, split screen, wipe etc.

In the early years, the methods were usually evaluated on relatively small datasets. Since 2001, the National Institute of Standards and Technology (NIST) has started a benchmark of content based video retrieval, i.e., TRECVID (Smeaton, et al. 2006), in which shot segmentation is included as one of the evaluation tasks. The practice of TRECVID tasks has significantly promoted the progress of shot segmentation techniques, revealing that the identification of abrupt cuts has been tackled to a satisfactory extent, however the detection of gradual transitions still remains a complicated and challenging problem.

The early work on shot detection mainly focused on abrupt cuts. In such approaches a cut is detected, when a certain difference measure between consecutive frames exceeds a threshold. The difference measure is computed either at a pixel or at a block level. Noticing the weakness of pixel difference methods, which is due to high sensitivity to object and camera motions, many researchers proposed the use of alternative measures based on global information, such as intensity and/or colour histograms (Patel and Sethi 1997), (Tsekeridou and Pitas 2001). Since then, the standard colour histogram-based algorithm and its variations have been widely used for detecting abrupt cuts. While the use of more complex features, such as image edges or histograms or motion vectors (Huang and Liao 2001) improves the results and performs very well for abrupt cuts, it does not solve the problem of gradual transitions (A. Hanjalic 2002). In another work, the authors have presented mathematical characterisations for most common transition effects (Albanese, et al. 2004), while more recent approaches deal with gradual transition detection using colour coherence change (Tsamoura, et al. 2008).

#### **2.1.1.2. Keyframe extraction**

As keyframe we consider the representative frame of an entire shot. In general, two applications of keyframe extraction can be considered: a) video indexing, browsing and retrieval and b) video summarisation and representation. The requirements for keyframes are to maintain the important content of the video

and remove any redundancy. To this end, the most reasonable approach would be to identify interesting objects, actions and events. However, due to the fact that so detailed semantic analysis is not currently feasible, the current works mostly rely on low-level image features or temporal information to extract representative keyframes.

In simple and most straightforward approaches, the first, the last frames and the temporally middle frames of the shot are selected as keyframes, regardless of the complexity of visual content motion analysis. On the other hand, the more sophisticated approaches take into account visual content, motion analysis and shot activity (Zhuang, et al. 1998). However, these approaches in many cases fail to effectively capture the major visual content or they are computationally expensive. In (Hanjalic and Zhang 1999) the frames of the shot are clustered and the centroids of the bigger clusters, also referred to as key clusters, are taken as keyframes. In another approach, discontinuity in the motion vectors provides information for the keyframes (Divakaran, et al. 2001). More recently, in (Besiris, et al. 2007) a keyframe extraction method based on a minimal spanning tree graph is proposed, where each node is associated to a single frame of the shot and the principle of maximum spread is applied to identify the keyframes.

#### **2.1.1.3. Low-level features extraction**

Low-level visual features such as colour, edge, texture, motion and salient points form the basis of similarity-based queries in video retrieval. Colour is considered as one of the most widely used features. The texture features characterise the different spatial patterns within the video, while the edge features may indirectly characterise the shape of the objects within the video. More recently the extraction of salient features from images and video keyframes has emerged.

The low-level visual features colour, texture and edge can be globally defined for a single keyframe, locally (spatially) defined for regular sub-regions of the keyframe, or even spatio-temporally defined for a sequence of video frames. Global representation is the simplest and the most common use of low-level features.

For the majority of video retrieval approaches the temporal domain does not play important role. Visual features are often defined in terms of a distribution across the image or the video segment, which are quite naturally represented as histograms. More generally, features including histograms can be physically represented as a vector describing the visual content and defining the similarity between queries and documents based on their distance. In a similar way, the extraction of salient points is usually exploited with the aid of the bag of words model, in which each salient point is associated with a cluster and finally the feature vector is the histogram of the distribution of the salient points in the identified clusters. In several approaches multiple features are combined into a single vector representation for the visual segment, which is often referred to as early fusion of visual features, and the vector may be further processed before calculating distances by applying normalisation (Hauptmann, et al., 2003) or dimensionality reduction (Nikolopoulos, et al., 2010).

Since 1999, the highly discriminative Scale-Invariant Feature Transform (SIFT) descriptor (Lowe 1999) is considered as one of the most popular descriptors in computer vision. Other examples of feature descriptors are Gradient Location and Orientation Histogram (GLOH) (Mikolajczyk and Schmid 2005), Speeded Up Robust Features (SURF) (Bay, et al. 2008), the machine-optimised gradient-based descriptors (Winder and Brown 2007), (Winder, et al. 2009) and the well established MPEG-7 descriptors (Manjunath, et al. 2002).

During the retrieval phase, both the query image and video keyframes have the same grid applied and matching is performed for the whole image. Alternatively, each regular region of the keyframe can be treated as a sub-image and shots can be ranked based on the best sub-image that matches the query image.

#### **2.1.1.4. High-level concepts and events**

One of the important challenges and perhaps the most interesting problem in semantic understanding of multimedia is visual concept detection. Many researchers have attempted to classify images as a whole, but the granularity is often too coarse to be directly used in real world applications. A well known set of high-level semantic concepts has been explored by the Large Scale Ontology

for Multimedia (LSCOM) initiative (Naphade, et al. 2006), a subset of which is used within TRECVID to study concept-based retrieval. Most approaches (e.g. (Gkalelis, et al. 2011), (Huiskes, et al. 2010)) in this area consider a supervised machine learning framework in order to train models that could predict the existence of a concept in a shot by exploiting annotated examples after extracting visual and motion low-level features.

More recently, significant research has been devoted to the detection and recognition of events in multimedia, in different application domains. Event detection can be considered even more challenging compared to concept identification, since events usually span along several video shots and can be decomposed into a variety of concepts under motion. For instance, “dog” could be a characteristic example of a visual concept, while “a human walking next to a barking dog” could be considered as an event. In this context, in (Xu and Chang 2008) a Bag-of-Words (BoW) algorithm is combined with a multilevel sub-clip pyramid method in order to represent a video clip in the temporal domain and the earth mover’s distance (EMD) is then used for recognizing events. More recently, Ballan, et al. (2010) present an algorithm that exploits knowledge embedded into ontologies and train SVM-based concept detectors to recognise events in the domains of broadcast news and surveillance. In (Jiang, et al. 2010), three types of features, namely, spatial-temporal interest points, SIFT features, and a bag of MFCC audio words, are used to train SVM-based classifiers for recognizing events. Finally, in (Moumtzidou, et al. 2010), a method for visual-only event detection in multimedia is presented, which is based on using a large number of pre-existing visual concept detectors for generating model vectors. Then, a combination of a dimensionality reduction technique and a nearest neighbour classifier based on the Hausdorff distance are applied to the model vectors, for associating videos with high-level events.

#### **2.1.1.5. Textual metadata extraction**

There are three possible text sources available in video retrieval: a) the text, which results after applying automatic speech recognition (ASR) on the audio part of the video, b) the text extracted with optical character recognition text (OCR) on the video frames and c) the closed caption (CC) text. The ASR text

represents the transcripts (i.e. what is spoken) of the audio part of the video. The video OCR is visible during the video and is commonly used in interviews and news reports to identify title, location, etc. The CC text is a representation or translation of the audio that is transmitted. In most of the times the later is not a word-by-word transcription of what is spoken and it includes change of speaker information and identification of audio events for specific programmes (e.g. knock at door, phone rings). These three different sources of text can be considered complementary.

The standard text processing that is applied to the ASR, video OCR and CC text is stopword removal, stemming and indexing. The textual information is easily aligned to shots using the available timestamps. In the case that the CC timestamps are missing, then they may be aligned with the video based on the ASR transcript (Rautiainen, et al., 2005). It is interesting to mention that despite the fact that the textual resources are synchronised with the video, there is no guarantee that the items reported in the text are visible in the associated shots. Finally there are cases, in which the textual information needs to be indexed in a different language from the initial one. To cope with this issue, automatic Machine Translation (MT) is employed, which however introduces noise to the text. This was the case in several years in TRECVID video search tasks, where the ASR text of Dutch videos has been automatically translated in English (e.g. (Zhang, et al. 2008)).

### **2.1.2. Video retrieval**

This section provides an insight in the video retrieval functionalities with a special focus on relevance feedback. Then, we present the interfaces used for video retrieval and finally we discuss the evaluation metrics employed for video search.

#### **2.1.2.1. Retrieval functionalities**

Since video data consist of heterogeneous sources of information, including text, audio visual features and generated metadata such as visual concepts, there are several ways, in which a user can formulate a query in a video retrieval system. As discussed in (Snoek, et al. 2007), three main query formulation paradigms

exist in the video retrieval domain: a) query by textual keyword, b) query by visual example and c) query by concept. In addition to these initial queries, the relevance feedback-based options, which involve the user in the search loop, can be considered as important functionalities of interactive retrieval systems. Since relevance feedback is very relevant to this thesis, we discuss it in detail in the next subsection.

The query by textual keyword is one of the most popular methods of searching for video (Hauptmann 2005). This query type is very simple and users are already familiar with this paradigm, since it is adopted from the traditional text-based searches. Query by text relies upon the availability of sufficient textual descriptions and annotations, including descriptive data and transcripts.

The query by visual-example is inspired by content-based image retrieval. This query type allows for the users to provide an image or a video shot as a visual example and retrieve similar results. This approach is based on the comparison of low-level features such as colour, texture, shape and salient points and could work satisfactorily for retrieving near duplicate images and keyframes. However, the main problem of content-based retrieval is that users in several cases expect not only visually similar results, but also semantically similar. Specifically, the subjectivity of human perception (Rui, et al. 1998) has as result that different persons (or even the same person under different conditions) may interpret visual content in a different way. It should be also mentioned that this functionality is becoming more popular in the recent years after being adopted by very well known search engines such as Google<sup>2</sup> and Yahoo!<sup>3</sup>.

More recently, a great deal of interest in the multimedia retrieval research community has been invested in query by concept (or event), which is also referred to as concept or event-based retrieval. Concept retrieval relies on semantic annotations, i.e. high-level concepts or events that have been associated

---

<sup>2</sup> <http://images.google.com/>

<sup>3</sup> <http://images.search.yahoo.com/>

with the video data (see 2.1.1.4). Assuming that semantic concepts can be considered as additional textual annotation, video documents can be retrieved by formulating textual search queries. In this context, query by concept can be considered as an extension to both query by textual keyword and query by visual example, since it includes textual input and considers visual features for performing retrieval.

#### **2.1.2.2. Relevance feedback**

Besides the aforementioned query options, several search engines provide a relevance feedback functionality to the users. In this case the user is capable of marking as relevant or non-relevant intermediate results and then the system provides an improved set of results. Relevance feedback techniques are usually based on supervised machine learning and the examples provided by the users are used as training samples.

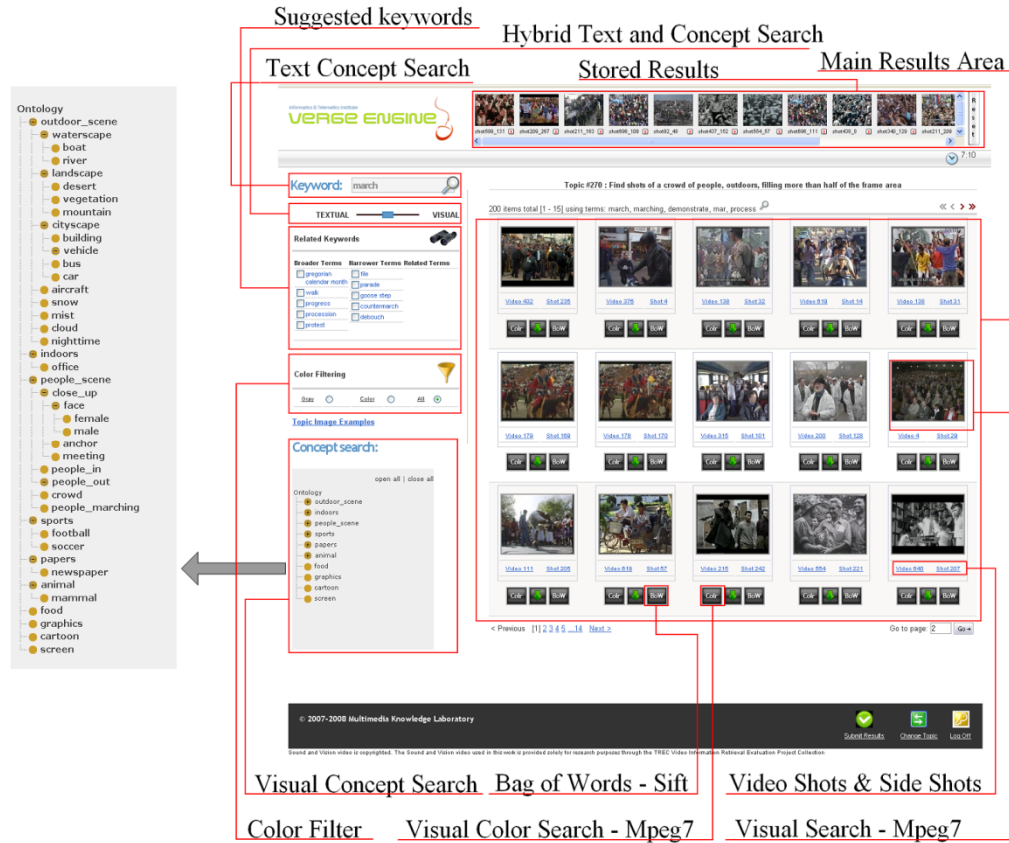
Historically, relevance feedback systems use machine learning techniques like expectation-maximisation (EM) and K nearest neighbours (KNN) to bring semantically similar results in response of any query image (Tao, et al. 2006). Other relevance feedback learning methods are based on Support Vector Machines (SVM) (Yildizer, et al. 2012), (Tong and Chang 2001) and Bayesian inference (Su, et al. 2011). SVM based methods consider the retrieval process as classification problem, in which relevant and irrelevant images are considered as training set. In an active learning approach (Yildizer, et al. 2012) the system selects the samples that fall near the SVM hyperplane and prompts the user to provide feedback. Constrained similar measure-based support vector machines (CSVM) (Azim-Sadjadi, et al. 2009) consider the images belonging to two clusters, construct a boundary to separate them and finally return sorted results.

Although relevance feedback functionality certainly improves the initial results of a video or image retrieval systems, it is not considered very popular due to the unwillingness of users to provide explicit feedback.

#### **2.1.2.3. Retrieval interfaces**

Graphical user interfaces of video retrieval systems serve as a mediator between the user and the video collection. These interfaces facilitate the users in

formulating search queries, retrieving results and browsing the video content. A detailed survey on representative video browsing and exploration interfaces is presented in (Schoeffmann, et al. 2010). In general, video retrieval interfaces can be divided into shot-based and story-based.



**Figure 2.2. VERGE video search engine**

In one of the first efforts for developing shot-based video retrieval interfaces, (Arman, et al. 1994) proposed to utilise the concept of keyframes (i.e. the representative frames of shots), for browsing the content of a video sequence in a temporal manner. Several other works have been published that are based on keyframe browsing of shots in a video sequence, usually by showing a page-based grid-like visualisation of keyframes (e.g. (Sull, et al. 2001), (Geisler, et al. 2002)). In another work (Adcock, et al. 2008) the authors present an interactive video search system called Media-Magic, which allows for searching at textual, visual, and semantic level, while in (Hopfgartner, et al. 2009) a tool for performing simultaneous search tasks within a video is proposed. Recently



VERGE (Vrochidis, et al. 2010c) video shot-based search engine (Figure 2.2) has been presented, integrating visual similarity search, concept-based retrieval and manually assisted linear fusion of heterogeneous modalities. Considering the large number of systems that visualise search results in a shot-based view, this approach can be seen as the standard visualisation method.

In some cases, shot-based retrieval is not considered as the ideal choice. To this end, story-based video retrieval interfaces have been devised. These interfaces usually consider news stories as the basic retrieval item. Examples of such search engines are NewsFlash (Haggerty 2004), which supports full text search and profile search, a similar system introduced in (Morrison S. Jose 2004), in which the web interface support query-by-textual-keyword, as well as NewsRoom proposed in (Diriye 2010). Consequently, not many system implementations exist for this scenario.

An additional reason that the shot-based interface prevails is the influence of the TRECVID evaluation campaign on video retrieval research, the tasks of which are focused on the video shot. However it should be mentioned that in the recent years TRECVID has introduced search tasks (i.e. known-item search), which are video-oriented and do not focus on shots.

#### **2.1.2.4. Video retrieval evaluation**

The information retrieval systems, approaches and techniques are usually evaluated by considering their effectiveness and computational efficiency. The system effectiveness depends mostly on two features: a) the ability of the system to model relevance, (i.e. to correctly associate documents to a given query) and b) the results presentation on a graphical user interface.

The majority of IR experiments focus on evaluating the system effectiveness. In this system-centred evaluation scheme, the system effectiveness is assessed, using well-established evaluation metrics, by comparing the output of the system with respect to a ground truth, which is manually constructed. On the other hand, in order to evaluate the presentation of the results, different interface models and their usability, a user-centred evaluation scheme is required. This evaluation scheme, which has its foundations in the area of human-computer interaction,

relies upon the explicit feedback of users to evaluate the system effectiveness and usability.

The system centred evaluation is considered as the most common assessment scheme in the IR community (Ingwersen 2005). The most well known evaluation metrics in IR are recall and precision, while average precision and F-score that combine the aforementioned metrics are also used.

Precision measures the proportion of the retrieved relevant documents. Based on the fact that users often interact with few results only, the top retrieved results can be considered as the most important ones. Assuming that during a search the system has retrieved  $M$  documents and the relevant retrieved results are  $M_R$ , the precision  $P$  is calculated as follows:

$$P = \frac{M_R}{M} \quad (2.1)$$

An alternative to evaluate these results is to measure the precision of the top  $N$  results. The  $P@N$  metric focuses on the quality of the top results, with a lower consideration on the quality of the recall of the system.

The recall measures the proportion of relevant documents that are retrieved in response to a given query. Assuming that during a search the system has retrieved  $M_R$  relevant documents, while the total correct documents to this query in the collection were  $M_C$ , the recall  $R$  is calculated as follows:

$$R = \frac{M_R}{M_C} \quad (2.2)$$

Both precision and recall are single-value metrics that consider the full list of the retrieved documents, while the ranking is not taken into account. Given the fact that most retrieval systems return a ranked list of documents, evaluation metrics should allow to measure the effectiveness of this ranking. One approach to combine these metrics is to plot precision versus recall in a curve.

Another popular measure of ranked retrieval runs is the “average precision” AP. Assuming that  $M_R$  are the relevant retrieved documents, and  $P_k$  the precision at rank of  $k$ -th relevant document, the AP is calculated as:

$$AP = \frac{1}{M_R} \sum_{k=1}^{M_R} P_k \quad (2.3)$$

In several cases the arithmetic mean of average precision AP over all queries, the “mean average precision” MAP is employed. MAP assumes that all queries are considered equal.

Finally, another popular evaluation measure that combines recall and precision is the F-score (also  $F_1$ -score or F-measure). It considers both the precision  $P$  and the recall  $R$  of the system to compute the evaluation metric. The F-score can be interpreted as a weighted average of the precision and recall, where an F-score reaches its best value at 1 and worst score at 0. The F-score ( $F_1$  score) is the harmonic mean of precision and recall:

$$F - score = \frac{2 \cdot P \cdot R}{P + R} \quad (2.4)$$

The other dimension of evaluation in search engines is the user-centred evaluation. In several cases there has been an open question in the research community regarding the application of system-centred or user-centred evaluation scheme in IR. For instance the authors in (Hancock-Beaulieu 1992) argue that system-centred evaluation is not suitable for interactive IR systems, since the controlled evaluation environment ignores essential factors of human-computer interactivity. In addition, they do not agree with the idea of using pre-defined relevance lists as ground truth. Aiming to address the main critique points towards the disadvantages of interactivity in the system-centred evaluation scheme, in (Borlund 2003) a framework for the evaluation of interactive information retrieval systems is introduced. The author argues that interactive IR systems should be evaluated under realistic conditions and suggests recruiting potential users as test subjects of the IR systems.

However, the recruitment of human subjects imposes several subjectivity parameters to the evaluation. Specifically, according to (Stanley and Campbell 1963), a well-known problem in such evaluations is the humans' learning aptitude. In other words humans learn how to handle a system better the longer they use it. Therefore, the results of subsequent experiments are most likely to be better than the results of early experiments. In addition, the involved users might become familiar with a specific topic and return better results than inexperienced users without any background knowledge. To this end, an approach is to average the results by several users, who are searching for the same topics, considering different topic search sequences (e.g. user 1 searches first for topic A and then for B, while user 2 searches first for topic B and then for A). Another well-established evaluation pattern to address this problem is called Latin-Square evaluation design, in which user and topic are treated as blocking factors. Assuming that we have two system variants, two topics and two users, the 2x2 latin square design (Figure 2.3) would be described by the following actions. Searcher S1 uses system variant V1 to search topic T1 and then system variant V2 to search topic T2, while Searcher S2 uses system variant V2 to search topic T1 and then system variant V1 to search topic T2.

	T1	T2
S1	V1	V2
S2	V2	V1

**Figure 2.3. 2x2 latin square design, in which two users are searching for two topics with two system variants**

## 2.2. Implicit user feedback in information retrieval

Differentiating from the traditional relevance feedback methods, which explicitly request the user to rate results, the use of implicit feedback techniques helps learning users' interest unobtrusively. The main advantage of this approach is considered the fact that the user is not prompted to provide explicit feedback. Since a large quantity of implicit data can be gathered without disturbing the user actions during retrieval, the implicit feedback-based approaches seem as an

attractive alternative. Of course not all the implicit user actions can be effectively interpreted or associated with relevance in information retrieval tasks. In this context, extended research has been performed to detect the features, which are valid indicators of interest.

In general case retrieval tasks, the implicit user feedback can be divided into two main categories: the query actions and the involuntary physical user reactions. The first category includes the patterns of user interaction with the search engine, such as series of mouse movements and clicks, shot selections, key strokes and keyboard inputs, while the second includes physical user unconscious behaviour, such as eye movements e.g. (Zhang, et al. 2010), heart rate and brain neuron reactions that can be gathered with electroencephalography (EEG) (e.g. (Arapakis, et al. 2009)). On the one hand, the feedback of the first category is easily gathered even during a web search session by using log-files, while physical reactions are recorded with the aid of special wearable devices or other sensors (e.g. cameras) capturing and analysing user behaviour. In the following sections we discuss in detail the relevant research performed in both of the aforementioned categories. First we present the works that deal with user navigation patterns in information seeking and then we report the research dealing with the implicit user feedback of the second category. Specifically, we focus on the role of gaze movements in information retrieval tasks, since this area is very relevant to this thesis. Finally, we also present techniques of exploiting user additional forms of implicit feedback in information retrieval, such as neuron reactions, which are used both for cognitive and affective-based tagging purposes.

### **2.2.1. Exploitation of past user interaction in information retrieval**

The study of past user interaction has started in the traditional text-based search engines and in the recent years the developed techniques have been transferred in the multimedia search systems. Therefore, we first provide an insight to the approaches applied to the textual search engines and then we discuss the works in multimedia retrieval.

#### **2.2.1.1. Exploitation of past user interaction in textual retrieval**

Implicit feedback approaches based on the user interaction with search engines have been effective in the context of textual retrieval, where they are mostly employed for query expansion and user profiling in order to retrieve, filter and recommend items of interest (Kelly and Teevan 2003).

The first works were aiming at quantifying the importance of implicit user feedback and investigate whether it could be exploited for retrieval. In this context, the authors in (Claypool, et al. 2001) introduce the definition of “Implicit Interest Indicators” by proposing specific user actions or combinations of them that can be considered as meaningful implicit feedback. In (White, Ruthven and Jose 2002), the authors perform a comparison between an explicit and an implicit feedback system concluding that there are not significant differences between them and that substituting the former with the latter could be feasible. In another work (Shinoda 1994), the authors evaluate whether user behaviour, while reading newsgroup articles, could be used as implicit indicator for interest. They measure the copying, saving or following-up of an entry and the time spent for reading the entries. They reveal that the reading time for documents rated as interesting is longer than for non interesting documents. However, a relation between interest and following-up, saving or copying has not been found.

In the context of further exploiting implicit feedback, the authors in (Seo 2000) introduce a method to learn users’ preferences from unobtrusively observing their web-browsing behaviour. Based on their conclusion the proposed approach can improve the retrieval performance. However, the adaptation of users’ interest over a longer period of time has not been taken into account as their search sessions have been set up only for a short period.

In another interesting work, the query-logs of a search engine are utilised to learn retrieval functions with the aid of machine learning (Joachims 2002). More specifically, the click-through data are translated into ranking user preference and then they are used to train a retrieval function with a Support Vector Machine (SVM) approach. The implemented SVMs in this case have been specifically designed in order to be trained by rankings, which declare related

preferences (i.e. one option is better than another). This development is considered more efficient and suitable for dealing with implicit feedback information, when compared to a typical SVM implementation that has to be trained with negative and positive samples.

In (Radlinski and Joachims 2005) the authors propose to detect query chains (i.e. a sequence of queries) and then learn a retrieval function using SVMs. The authors demonstrate a simple method for automatically detecting query chains in query and click-through logs. These data are used to infer preference judgments regarding the relative relevance of documents both within individual query results, and between documents returned by different queries within the same query chain. The method used to generate the preference judgments is validated using a controlled user study. A ranking SVM is adapted to learn a ranked retrieval function from the preference judgments. The results demonstrate significant improvements in the ranking given by a normal search engine.

More recent works focus on evaluating different ranking algorithms with the aid of implicit information (i.e. user click selection), either by comparing different ranking functions or by merging results of different algorithms (Radlinski, et al. 2008).

#### **2.2.1.2. Past user interaction-based approaches in multimedia retrieval**

Implicit feedback techniques have not been fully explored in the multimedia domain (Hopfgartner and Jose 2007). In textual retrieval, the usual implicit information that can be taken into account is the user selection (i.e. the user clicks on an interesting link or textual description to view the complete document), while in video retrieval there are multiple interactions between the user and the system, defining in that way many implicit indicators (e.g. submission of textual, visual or temporal queries, etc).

The main approach to exploit the user feedback during video retrieval interactive sessions is to extend the idea of “query chains” (Radlinski and Joachims 2005) and construct a graph that describes the user action. Subsequently this graph is transformed into a weighted graph by aggregating the links between the same nodes and weights are introduced based on the different actions taken into

account. In that way, links between the data and the submitted queries are initiated. Recent works employ the aforementioned technique to deal with user clicks.

More specifically in (Hopfgartner, et al. 2008) the authors propose to use community based feedback mined from the interactions of previous users of a video retrieval system, which is based on Okapi BM25 retrieval model supporting text queries to aid users in their search tasks. This feedback is the basis for providing recommendations to new users of the video retrieval system. This is performed by representing all user interactions with a weighted graph. Then, this implicit information is aggregated from multiple sessions and users into a single representation, thus facilitating the analysis and exploitation of past implicit information. In (Vallet, et al. 2008) the authors evaluate 4 different algorithms that can be applied on the weighted graph to provide recommendations. The evaluation is performed with simulated users, whose navigation action is based on a statistical behaviour of real users. The results of these works seem to be promising as they complement the existing baseline text and relevance feedback systems.

In another approach (Craswell and Szummer 2007), the authors apply a Markov random walk model to a large click log, producing a probabilistic ranking of documents for a given query in an image search engine. The model is able to retrieve relevant documents that have not yet been clicked for that query and rank those effectively. They conduct experiments on clicked logs during image search, comparing the proposed ('backward') random walk model to a different ('forward') random walk, reporting that the most effective combination is a long backward walk with high self-transition probability.

In (Yang, et al. 2007), a video retrieval system is presented, which employs relevance feedback and multimodal fusion of different sources (textual, visual and click-through data), in order to generate recommendations for the user. In this approach, the textual, visual and aural data of the video shots are processed separately and compared with the selected video document. Then, these results are fused. A further adjustment of the fusion weights is performed with the aid of the click through data, which denote the interest of the user to a specific



document based on the time she/he watched the video shot. The approach of fusing content analysis information with the implicit feedback seems to be very interesting and promising, however in this specific work the implicit information is not very deeply exploited, as the sequence of query actions is not taken into account, failing in that way to semantically interconnect subsequent queries and shots.

A similar approach is proposed in (Moumtzidou, et al. 2011), in which the time duration of a user hovering on a shot during video retrieval tasks is considered as the main implicit interest indicator. The authors are based on the assumption that there are topics for which specific visual concepts and metadata are important (or are more descriptive than others). With a view to exploiting this assumption, the implicit feedback information is utilised, in order to train weights between different modalities or between instances of the same modality, which are used by a fusion function.

Overall it seems that most of the state of the art works consider only textual queries, while basic video retrieval options as query by visual example and temporal queries are not taken into account. In addition, fusion or combination of aggregated click-through data with the content-based modalities is not attempted.

### **2.2.2. Eye movement-based approaches**

Studies utilising eye movements in order to investigate cognitive processes started to appear three decades ago. Based on this research, eye movement data, which are categorised in: fixations, saccades, pupil dilation and scan paths, have proven to be very valuable in studying information processing tasks (Rayner 1998). Eye tracking methods are mostly used in information retrieval tasks in order to identify items of interest, as well as to understand the behaviour of the user.

#### **2.2.2.1. Eye movement-based works in textual retrieval**

Most of the works that employ gaze analysis in the area of information retrieval focus on textual document search.

In such a work (Granka, et al. 2004), the authors investigate how the users interact with the results of a web search engine by employing eye-tracking. A very interesting approach is described in (Puolamaki, et al. 2005), in which proactive information retrieval is proposed by combining implicit relevance feedback and collaborative filtering. More specifically, implicit feedback is inferred from eye movement signals, with discriminative Hidden Markov Models (HMM) estimated from data, for which explicit relevance feedback is available. Eye movements are modelled with a two-level discriminative HMM, where the first level models transitions between sentences, whereas the second level models transitions between words within a sentence. Collaborative filtering is carried out using the User Rating Profile model.

Other approaches attempt to evaluate and interpret meaningfully the user behaviour during text retrieval tasks (Joachims, et al. 2005). In this case, the gaze movements are used to model and understand the user behaviour and decision process and finally propose strategies to generate feedback from clicks. Focussing only on gaze fixations, the authors conclude that results are usually viewed from top to bottom and that the lower a click in the ranking, the more abstracts are viewed above the click. In another work (Brooks, et al. 2006), restructuring of the information that is presented to the user during a text retrieval task based on eye-tracking is proposed. The measures considered are: cardinality of fixations, fixation duration, pupil size, and regressions. In a more recent work (Kirkegaard Moe, et al. 2007), the authors attempt to identify indicators and features for eye-tracking in text retrieval considering viewing time, thorough reading and regressions.

In this context, the authors in (Hardoon, et al. 2007) introduce a search strategy, in which a query is inferred from information extracted either from eye movements measured when the user is reading text during an information retrieval (IR) task or from a combination of eye movements and explicit relevance feedback. A SVM implementation is employed both for predicting relevance between unseen documents and for combining eye movement and textual features.

In another work (Ajanki, et al. 2009), an implicit information retrieval query is inferred from eye movements measured when the user is reading, and used for query expansion. During the training phase, the user's interest is known, and a mapping is learned from how the user looks at a term to the role of the term in the implicit query. Then, this mapping is used to construct queries even for new topics, for which no learning data are available.

Finally, in another approach (Buscher, et al. 2008), the authors present a method for discriminating skimming from reading.

#### **2.2.2.2. Eye movement-based approaches in multimedia retrieval**

The exploitation of eye movements in multimedia search followed the common practice in information retrieval of reusing and extending the work proposed for textual search. The first applications of eye-tracking in image and video retrieval were in the area of studying the user behaviour and evaluating interface representations. Then, the research trend moved to more challenging problems such as the identification of user interest and automatic annotation.

More specifically, in (Hughes, et al. 2003) an eye-tracking study is conducted to investigate whether the textual or the visual representation of video is mostly considered by users in a search engine interface. Based on the results, the users seem to pay more attention to the textual information. In another work (Moraveji 2004), eye-tracking is applied to evaluate an approach, in which a video timeline is enriched with colour information from the video visual data. Based on the gaze movements the authors conclude that their approach is indeed interesting for the user. Recently, the authors in (Castagnos, et al. 2010) perform an experiment using eye-tracking system in order to collect users' interaction behaviours as they browse and select products to buy from an online store. They consider the aggregated eye fixations of the users, in order to derive, which parts of the interface of a recommender system are of interest.

More recent works in image and video retrieval deal with deriving user interest based on eye movements (focusing mostly on fixation and saccades) and also utilise this technique to develop gaze-based interactive interfaces. In (Oyekoya and Stentiford 2004a) and (Oyekoya and Stentiford 2006) the idea of an

interactive interface for image retrieval is proposed. In these interfaces the input is provided by the eye movements of the user, concluding that eye-trackers could support such an implementation. Furthermore, in (Kozma, et al. 2009) the real time interface GaZIR for browsing and searching images is proposed. In this approach, the relevance of the viewed images is predicted by considering fixation and saccade-based features, while relevance prediction is performed with classical logic regression. In a similar application (Santella, et al. 2006) fixation features are used to identify important content and perform photo cropping.

In (Jaimes, et al. 2001) the authors explore the way, in which people look at images of different semantic categories (e.g., handshake, landscape), and attempt to perform automatic image classification. In this context, they conduct eye-tracking experiments, which show that similar viewing patterns occur when different subjects view different images in the same semantic category. They propose a system, in which image classifiers are trained using machine learning from user input as the user defines a multiple level object definition hierarchy based on an object and its parts, and labels examples for specific classes (e.g., handshake). The authors also investigate the use of fixations, in order to automatically select the important regions of the images during the training phase.

Other works in image and video retrieval attempt to derive user interest based on eye movements. In (Oyekoya and Stentiford 2004b) the authors conduct experiments to explore the relationship between gaze behaviour and a visual attention model that identifies regions of interest in image data. The reported results based on analysis of the fixation duration show that there is a difference in behaviour on images depending on whether they contain a clear region of interest.

The authors in (Ramanathan, et al. 2009) propose a framework to localise and label affective objects and actions in images by combining text, visual and gaze-based analysis. The affect model is derived from fixation patterns on labelled images, and guides localisation of affective objects (faces, reptiles) and actions (look, read) from fixations in unlabeled images.

In another work (Klami, et al. 2008), the authors propose nine-feature vectors from different forms of fixations and saccades and use a classifier to predict one

relevant image from four candidates in two steps: a) first they extract features from the eye trajectory and employ a binary classifier to determine whether a specific page includes images of interest and b) they extract features for each image and use a 4-class classifier to detect which image is of interest.

Recently, an approach for performing relevance feedback based on eye movements is proposed in (Zhang, et al. 2010). More specifically, this work employs eye-based features and the construction of a decision tree, which is trained using ground truth provided by the users.

In another recent work (Hajimirza and Izquierdo 2010), the authors attempt to automatically annotate images by exploiting the gaze movements of the user during daily surfing in the Internet or in visual database. Specifically, in the proposed framework two subsequent Fuzzy Inference Systems (FIS) are employed, in order to assign relevance values to the viewed images with respect to a given target concept. The first FIS classifies visit period and number of revisits, while the second FIS generates a Gravity Vector, which moves the relevance value of the previous users towards the relevance value of the current user. The preliminary results indicate that in a multi-user environment the annotating precision of the system is over 80% with the recall between 60%-80%.

In an extension of the previous work (HajiMirza, et al. 2011) the authors investigate using gaze movements as a form of feedback for media personalisation and adaptation. Descriptive features are extracted from the gaze trajectory of users, while they are searching in an image database. These features are used to measure a user's visual attention to every image appeared on the screen. For every new user a new adapted processing interface is developed automatically. The authors argue that the gaze movements comprise a reliable feedback to be used for measuring one's interest to images, which helps to personalise image annotation and retrieval.

Besides fixations and saccades, pupil dilation has been also studied as an indicator of user interest during visual detection tasks. An interesting work, which falls into the area of visual target detection, is proposed in (Privitera, et al. 2008). The authors investigate whether the pupil response can be considered as a

reliable marker of a visual detection event, while viewing complex imagery. After conducting experiments, in which viewers were asked to report the presence of a visual target during rapid serial visual presentation, the conclusion is that pupil dilation is significantly associated with target detection. In another work (Qian, et al. 2009), pupil information is used to improve the performance of an image classification system based only on EEG signal analysis. More specifically, pupil responses are proposed as a complementary modality and are utilised for feature-extraction. A two-level linear classifier is then used to obtain cognitive-task-related analysis of EEG and pupil responses.

Finally, more recent works in image retrieval attempt to combine image features with eye movements. In this context, the authors in (Hardoon and Pasupa 2010) propose a search methodology, which combines image features together with implicit feedback from users' eye movements in a tensor ranking Support Vector Machine and show that it is possible to extract the individual source-specific weight vectors. In addition, they demonstrate that the decomposed image weight vector is able to construct a new image-based semantic space that outperforms the retrieval accuracy than when solely using the image-features.

In (Liang, et al. 2010), the authors exploit the gaze information (fixations) to identify which part (i.e. region) of the image the user mostly looks to. Then, they perform image segmentation and extraction of local features. The parts of the image that seem to be mostly viewed are considered as most important and consequently greater weights are assigned to the local features extracted from these areas.

In (Faro, et al. 2010), an implicit relevance feedback method is proposed with a view to improving the performance of image retrieval systems by re-ranking the retrieved images according to users' eye gaze data. In detail, after the retrieval of the images by querying the image retrieval engine with a keyword, the proposed system computes the most salient regions (where users look with a greater interest) of the retrieved images by gathering data from an unobtrusive eye tracker. Subsequently, local features are extracted and reranking is performed based on similarity scores (i.e. distances based on local features) computed

between the relevant images identified by the eye tracker and the rest of the retrieved images.

In another work (Pasupa, et al. 2009) an image search strategy is presented, which combines image features together with implicit feedback from users' eye movements to rank images based on a perceptron formulation of the Ranking Support Vector Machine algorithm.

Finally, in (Walber, et al. 2012) the authors investigate the principle idea of identifying specific objects shown in images by looking only at the users' gaze path information. Specifically, for each image region the fixation measures are calculated over all gaze paths and summed up per region. By analyzing the gaze paths, a 67% of the image regions is correctly identified.

Overall, although research has been conducted towards gaze movement-based feature extraction, the existing techniques do not consider early or late fusion of fixation and pupil dilation-based features. In addition, most of the approaches do not consider aggregated gaze-movement data and the experiments are performed in strictly controlled environments (i.e. the users are instructed to look at interesting images).

### **2.2.3. Additional implicit feedback approaches in multimedia retrieval**

Besides the gaze movements, several other modalities of implicit feedback have been used for gaining information of an unknown multimedia dataset. Such modalities have been used either to achieve cognitive implicit tagging and identify user interest or to extract affective information.

#### **2.2.3.1. Cognitive implicit tagging**

Implicit tagging research has recently attracted researchers' attention, and a number of studies have been published, most of them based on recording the brain response with electroencephalography (EEG). The use of EEG in this process is interesting mainly because it offers the possibility of passive, implicit tagging. This means that tags are generated by analysing the EEG data as subjects consume multimedia data, without active involvement or conscious

effort on their part. Implicit tagging is defined as using non-verbal behaviour to find relevant keyword or tags for multimedia content (Pantic and Vinciarelli 2009). While at the moment the recording of EEG measurements is still a quite cumbersome process, the recent improvements in the development of dry electrodes may simplify the use of this modality and make it usable outside of the laboratory environment. The employment of EEG in annotating multimedia data is a rather recent research direction and so far only a few works have investigated this area.

In this context, the authors in (Gerson, et al. 2006), consider a paradigm, in which images of a forest environment are shown to subjects for 100 msec each. The goal is to detect a small subset of target images that contained pedestrians. The target images elicit a P300 event-related potential (ERP) (i.e. the measured brain response that is the direct result of a specific sensory, cognitive, or motor event) (Luck 2005), which is then classified using Fisher linear discriminant analysis. Another test is run without considering the EEG modality, where subjects are instructed to press a button upon seeing the target images. The results show no significant differences in target image detection accuracy between the use of the EEG modality and the use of buttons.

In another work (Kapoor, et al. 2008), categories of images are classified based on EEG measurements recorded during the presentation of images. The categories employed are faces, animals and inanimate objects. This is based on the notion that the human visual system responds very differently to images that fall into the aforementioned categories. The authors propose a vision-based algorithm that uses pyramid match kernels to initially classify the images. Then, the EEG data are combined with the vision-based features using a kernel-alignment method. Based on the evaluation the combination of the two modalities outperforms the individual methods.

In (Cowell, et al. 2008) the authors use ERP analysis in combination with eye-tracking to assist intelligence analysts in rapidly reviewing and categorizing satellite imagery. The analyst is assigned a target category to look for in the images. When subjects see an image in the target category, an ERP occurs in the



EEG data, which is then classified. The gaze movements are used to determine points of interest within the images.

More recently in (Koelstra, et al. 2009), the authors attempt to find neuro-physiological indicators to validate tags attached to video content. Subjects are shown a video and a tag and they aim to determine whether the shown tag is congruent with the presented video by detecting the occurrence of an N400 event-related potential. The idea is that tag validation could be used in conjunction with a vision-based recognition system as a feedback mechanism to improve the classification accuracy for multimedia indexing and retrieval.

#### **2.2.3.2. Affective-based tagging**

In addition to the aforementioned works, research has been devoted to achieve affective tagging of multimedia by taking into account physiological responses, facial expressions and brain neuron reactions. Affective tagging and retrieval deals with the extraction of emotion descriptive metadata.

In this context, the authors in (Kierkels, et al. 2009) propose a method for personalised affective tagging of multimedia using peripheral physiological signals. Valence and arousal levels of participants' emotion when watching videos are computed from physiological responses using linear regression. Then, quantised arousal and valence levels for a clip are mapped to emotion labels. This mapping enables the retrieval of video clips based on keyword queries. However, this novel method achieved low precision.

The authors in (Joho, et al. 2010), (Joho, et al. 2009) have developed a video summarisation tool based on facial expressions. In this approach, a probabilistic emotion recognition based on facial expressions is employed to detect emotions of 10 participants watching eight video clips. The participants are asked to mark the highlights of the video with an annotation tool after the experiments. The expression change rate between different emotional expressions and the pronounce level of expressed emotions are used as features to detect personal highlights in the videos. The pronounce levels employed range from highly expressive emotions, surprise and happiness, to no expression or neutral. In addition, two affective content-based features (audio energy and visual change

rate from videos) are extracted to create an affective curve in the same way as the affective highlighting method proposed in (A. Hanjalic 2005).

Then, the authors in (Arapakis 2009b) introduce a method to assess the topical relevance of videos in accordance to a given query using facial expressions showing users' satisfaction or dissatisfaction. Based on facial expressions recognition techniques, basic emotions are detected and compared with the ground truth. They are able to predict with 89% accuracy whether a video is indeed relevant to the query.

In a more recent study, the feasibility of using affective responses derived from both facial expressions and physiological signals as implicit indicators of topical relevance has been investigated. Although the results are above random level and support the feasibility of the approach, there is still room for improvement from the best obtained classification accuracy, 66%, on relevant versus non-relevant classification (Arapakis 2009a).

In another recent work (Yazdani, et al. 2009) the authors propose to use a Brain Computer Interface (BCI) based on P300 evoked potentials to emotionally tag videos with one of the six Ekman basic emotions (Ekman, et al. 1987). The proposed system is trained with 8 participants and then tested on 4 others. A high accuracy on selecting tags is achieved. However, in this system, the BCI only replaces the interface for explicit expression of emotional tags, which means that the method does not implicitly tag a multimedia item using the participant's behavioural and psycho-physiological responses.

Finally, in an approach (Koelsch, et al. 2004) that deals with music affective tagging, an N400 response is observed for labels presented after musical excerpts. These labels are attributed to the music in terms of associated objects (e.g. birds, needles), musical features, and moods. Given the fact emotions are subjective in nature, the N400 approach to tag validation introduced in this work could in principle assess the subjective response to media content.

## **Chapter 3**

### **VIDEO CONTENT ANALYSIS**

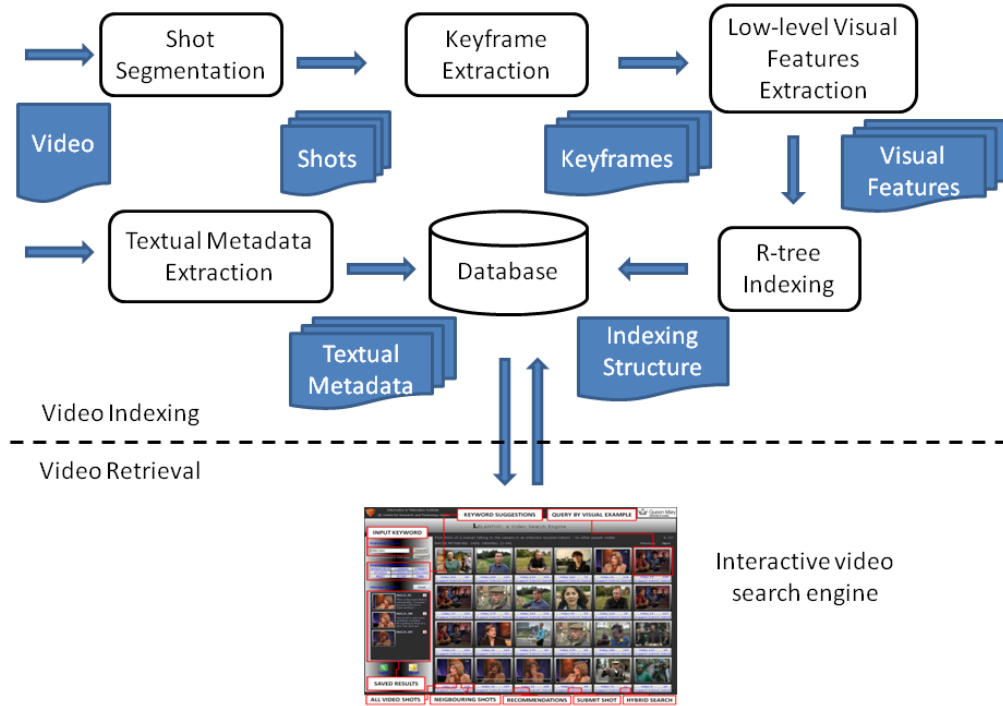
*This chapter discusses the video indexing and retrieval techniques we have applied for implementing of a video search engine with a view to conducting interactive video retrieval experiments. First, we present the employed video processing techniques including shot segmentation, keyframe extraction, as well as textual indexing and extraction of low-level visual features. Then we demonstrate the video search engine and the supported retrieval functionalities and we provide implementation insights.*

#### **3.1. Introduction**

As discussed in the previous chapter, to perform video retrieval, we need to index efficiently the multimedia dataset. Following the recent work in video indexing and retrieval, we have applied well established image and video processing techniques and developed an interactive video search engine, in order to conduct experiments and evaluate the proposed algorithms that exploit implicit user feedback. The framework we employed towards this goal is illustrated in Figure 3.1, and is based in the standard approach for video indexing and retrieval as this was presented in Figure 2.1.

The framework includes temporal indexing of the video source by performing shot segmentation and keyframe extraction. In the following, low-level MPEG-7 visual features are extracted to enable retrieval functionalities based on visual similarity. To ensure fast response of the search engine, an R-tree indexing structure is employed to index efficiently the low-level features in the multidimensional space. In addition, textual information from the audio part of

the video is extracted and indexed with textual indexing algorithms. The extracted information is stored in a relational database, which is accessed by a web-based video search engine at run-time.



**Figure 3.1. Video indexing and retrieval framework**

In the rest of the chapter we present a description of the dataset we have used and then we describe the algorithms we have applied for video indexing. Since in this chapter we employ well established techniques of video processing and analysis, the results are directly reported after the algorithm presentations, given the fact that no evaluation is expected at this stage. Finally, we present the implemented interactive video search engine.

### 3.2. Dataset

In this work we made use of the TRECVID 2008 test video set by NIST<sup>4</sup>, which

<sup>4</sup> National Institute of Standards and Technology (NIST): <http://www.nist.gov/>

consists of about 100 hours of Dutch video (news magazine, science news, news reports, documentaries, educational programming, and archival video). In Figure 3.2 we provide representative visual examples of the dataset. The video includes indoor and outdoor action, faces, humans, as well as colourful and black and white scenes. This set is also accompanied with annotated shots for 24 query topics, which were used in the search task of TRECVID 2008. Part of these query topics are also used in our experiments. The ground truth and the annotations for these topics are provided by NIST.

In addition, in order to train the video shot detection module (section 3.3.1.1), we have used 10 minutes segments of TRECVID 2007 dataset, which includes a variety of content such as news, documentaries and sports.



Figure 3.2. Example keyframes of the TRECVID 2008 test video set

### 3.3. Video indexing

Video indexing is performed in three dimensions. These include temporal, text-based and visual-based indexing.

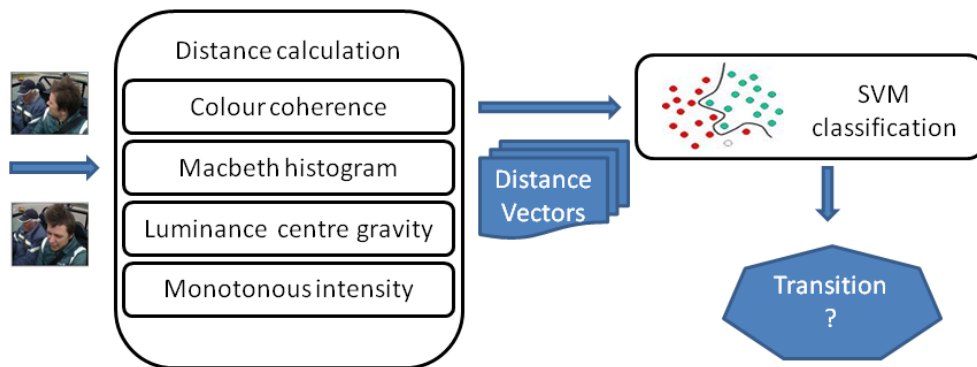
#### 3.3.1. Temporal indexing

In order to generate an efficient representation of the initial video source and index it according to temporal information, shot boundaries detection and shot segmentation steps are required to split the video into shots, which comprise the items to be retrieved. As already discussed (section 2.1.1), shot boundary detection provides the basis for almost all high-level video content analysis approaches, validating it as one of the major prerequisites for successful indexing and retrieval in large video databases.

### 3.3.1.1. Shot segmentation

In this work, shot detection is achieved by following the approach proposed in (Tsamoura, et al. 2008). This method was selected, since it has already been evaluated using the dataset of TRECVID 2007 (which covers thematic categories similar to TRECVID 2008 dataset) with satisfactory results (Tsamoura, et al. 2008).

This approach considers four individual criteria for gradual transition detection, which include colour coherence change, Macbeth colour histogram change, luminance centre of gravity change and monotonous intensity change. Then, these criteria are combined in a meta-segmentation scheme, in order to achieve more accurate detection results. In this schema the selected features are initially computed for all the video frames. Given a couple of consecutive frames, the distances between the above features are computed forming distance vectors. These vectors are subsequently supplied to a trained binary classifier, the output of which denotes whether the vector (and therefore the intermediate between the two frames it represents) is part of transition area or not. The shot segmentation framework is illustrated in Figure 3.3. In the following, we discuss in details the distance calculation and the classification steps.



**Figure 3.3. Shot segmentation framework**

Colour Coherence Vectors (CCV) have been proposed for applications that involve image retrieval (Pass, et al. 1996) to alleviate the drawback that colour histograms do not provide any information regarding the spatial arrangement of colours in the image. Colour coherence expresses the degree of colour's

accumulation in an image area. Coherent pixels belong to contiguous regions of size greater than  $\psi$ , in contrast to incoherent pixels. Before computing the coherence, we apply colour quantisation using the Macbeth colour pallet (McCamy, et al. 1976), which consists of 24 colours. Pixel colours are mapped to one of the 24 colours of the Macbeth pallet by constructing a 24 bins histogram. Let  $Z_i$  where  $i = 1..24$  denote the Macbeth pallet colour clusters. Then, each pixel  $p(x,y)=[R_{xy}G_{xy}B_{xy}]$  is assigned to the colour cluster  $Z_i$  for which the L1 distance between  $Z_i$  and  $p$  is minimised. Then, we classify the pixels of a given colour class as either coherent or incoherent. A coherent pixel is part of a connected spatial region, the pixels of which belong to the same colour class. A connected component  $C$  is a set of pixels such that for every couple of pixels  $p$  and  $p' \in C$ , there is a path in  $C$  between them. For each Macbeth colour cluster  $Z_i$   $i = 1 \dots 24$ , some of its pixels will be coherent, while the others will be incoherent. Let  $c_i$  be the number of coherent pixels of  $Z_i$  and  $d_i$  the number of incoherent pixels. The total number of pixels belonging to  $Z_i$  is  $c_i + d_i$ , resulting in a Macbeth color histogram:

$$M_t = [c_1^t + d_1^t \ c_2^t + d_2^t \ \dots \ c_{24}^t + d_{24}^t] \quad (3.1)$$

The colour coherence vector is then defined as:

$$G_t = [(c_1^t, d_1^t) \ (c_2^t, d_2^t) \ \dots \ (c_{24}^t, d_{24}^t)] \quad (3.2)$$

Based on (Tsamoura, et al. 2008) we have set  $\psi$ , the size of the smallest coherent area to 1% of the number of pixels in each frame. The distance between frames  $I_t$  and  $I_{t-1}$  having  $G_t$  and  $G_{t-1}$  colour coherence vectors respectively, is computed as:

$$D_t^G = \sum_{i=1}^{24} (|c_i^t - c_i^{t-1}| + |d_i^t - d_i^{t-1}|) \quad (3.3)$$

where  $t = 0, \dots, T$  and  $T$  corresponds to video duration.

The distance between frames  $I_t$  and  $I_{t-1}$  using their Macbeth colour  $M_t$  and  $M_{t-1}$  is estimated as described in the previous paragraph and can then be defined as:

$$D_t^M = \sum_{i=1}^{24} (|c_i^t - c_i^{t-1}| + |d_i^t - d_i^{t-1}|) \quad (3.4)$$

Computing the distances between pairs of consecutive frames based on the Macbeth colour histogram feature, a curve  $D_t^M$ ,  $t = 0, \dots, T$  is produced.

Luminance centre of gravity is defined in an analogous way to an object's centre of mass: it is the point where luminance is concentrated on. Let  $L_t(x, y)$  be the luminance image calculated for frame  $I_t$ . Then the luminance centre of gravity of the frame is computed as:

$$R = [R_x \ R_y] \quad (3.5)$$

$$R_x = \frac{\sum_x x L_t(x, y)}{\sum_x L_t(x, y)} \quad (3.6)$$

$$R_y = \frac{\sum_y y L_t(x, y)}{\sum_y L_t(x, y)} \quad (3.7)$$

The Euclidean distance of the luminance centres of gravity between frames  $I_t$  and  $I_{t-1}$ , having  $R_t$  and  $R_{t-1}$  respectively is:

$$D_t^R = ||R_t - R_{t-1}|| \quad (3.8)$$

Computing distances between frames based on the luminance centre of gravity feature for an input video, a  $D_t^R$  curve  $t = 0, \dots, T$  is produced.

The monotonous intensity change is described by the change of the percentage of pixels with monotonously varying intensities. If it exceeds a certain threshold, a dissolve/fade transition is detected. Let  $f(x, y, t) = L_{t+1}(x, y) - L_t(x, y)$ . Then, the monotonous change of intensity is evaluated using the following equation:

$$g(x, y, t) = \begin{cases} 1, & f(x, y, t)f(x, y, t-1) \geq 0 \\ 0, & f(x, y, t)f(x, y, t-1) < 0 \end{cases} \quad (3.9)$$

Subsequently, the percentage of pixels with monotonously varying intensities at a given time  $t$  can be calculated as  $D_t^I = \sum_{xy} g(x, y, t)$ .



Since a single criterion is difficult to accommodate for all possible effects that hinder gradual shot detection, a combination of multiple individual criteria is employed to improve detection accuracy. To this end, a machine-learning classification approach is adopted, based on SVM. For training, the classifier is supplied with a set of input vectors manually assigned to the appropriate class (transition or non-transition). Jointly considering all the aforementioned criteria would result in a 4-dimensional distance vector  $D_t$  between frames  $I_t$  and  $I_{t-1}$ :

$$D_t = [D_t^I D_t^M D_t^R D_t^G] \quad (3.10)$$

whereas using a subset of the criteria is also possible by defining a distance vector of lower dimensionality. For classification a C-SVM with a radial basis function kernel of 3<sup>rd</sup> degree is employed.

About 10 minutes segments of TRECVID 2007 test set is used for training. After applying this technique to the dataset described in section 0, the latter is segmented into 35766 shots.

### 3.3.1.2. Keyframe extraction

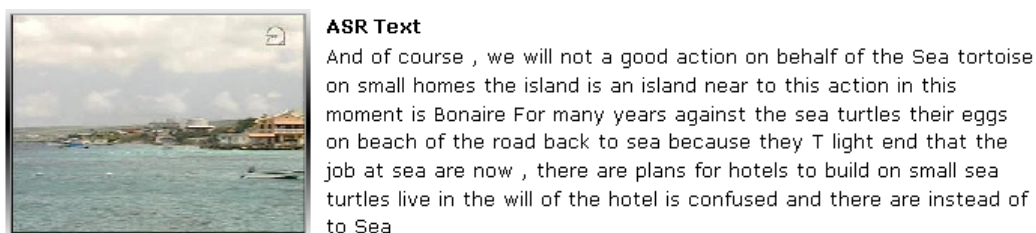
After we have performed the shot segmentation procedure, the representative keyframe for each shot has to be extracted. Based on the fact that complicated approaches (e.g. content-based techniques) are computationally expensive and do not always perform better than simplistic approaches (as discussed in section 2.1.1.2), we apply a straightforward technique to perform keyframe extraction. Specifically, in order to avoid high computational cost and achieve fast performance, we select the temporally middle frame as the representative one following the approach in (Moumtzidou, et al, 2009) and (Moumtzidou, et al. 2010).

Therefore, for the aforementioned dataset, we end up with 35766 representative keyframes (i.e. one keyframe per video shot). In that way a temporal indexing structure is constructed, and therefore each video can be represented by a sequence of images (i.e. one image per video shot).

### 3.3.2. Textual indexing

Indexing of video shots according to the associated textual information is realised using the Lemur (The Lemur Project n.d.) toolkit, which is one of the well known libraries for text retrieval and has been successfully applied in similar works on interactive video retrieval, such as in (Moumtzidou, et al. 2009, 2010 and 2011).

In this work, the audio information is processed off-line with the application of Automatic Speech Recognition (ASR) on the initial video source, so that specific sets of keywords can be assigned to each shot. Given the fact that the initial audio is provided in Dutch, a Machine Translation (MT) step from Dutch to English is performed. The ASR and the MT are offered by the University of Twente<sup>5</sup> in the context of TRECVID 2008. However, due to the errors that are usually employed during automatic speech recognition and machine translation they cannot be considered as highly reliable. This fact makes the video retrieval problem even more challenging, since the usually most reliable source of metadata (i.e. textual annotation from audio) is noisy. In Figure 3.4 we illustrate an example of a shot represented as a keyframe and the associated transcription. It is interesting to notice how noisy and unstructured the text transcripts become after the employment of ASR and MT. It should be noted that the ASR and MT, provide a direct association of the transcripts with the temporal line of the video.



**Figure 3.4. Example of shot and associated ASR**

---

<sup>5</sup> University of Twente: [www.utwente.nl/](http://www.utwente.nl/)

Based on the shot boundaries detection task implemented in the previous section, we are able to map each shot to a set of keywords. Then, we apply text indexing using tools provided by the Lemur project (The Lemur Project n.d.).

In order to clear the textual information from unwanted data we need to remove common uninformative words (e.g. the, at) and the word suffixes. To this end, we manually construct a list of unwanted stopwords, while stemming, which is the process for reducing inflected (or sometimes derived) words to their stem, base or root form, is performed by the application of the Porter stemming algorithm (Porter 1980).

Then, we index the textual information by employing the Indri search engine (Strohman, et al. 2004) from Lemur project. The retrieval model implemented in the Indri search engine is an enhanced version of the model described in (Metzler, et al. 2004), which combines the language modelling (Ponte 1998) and inference network (Turtle and Croft 1991) approaches to information retrieval. The resulting model allows structured queries to be evaluated using language modelling estimates within the network, rather than standard *tf.idf* estimates. The documents are represented as multisets of binary feature vectors. The features can be nearly any interesting binary observation of the underlying text. During indexing, the system builds compressed inverted lists for each term and field in memory.

When a query is submitted to the Indri system, it is evaluated in two phases. In the first phase, statistics about the number of times terms and phrases appear in the collection are gathered. In the second phase, the statistics from the first phase are used to evaluate the query against the collection. The documents are ranked according to  $P(\frac{I}{D}, \alpha, \beta)$ , which stands for the belief that the information need  $I$  is met giving document  $D$  and hyperparameters  $\alpha$  and  $\beta$  as evidence.

### 3.3.3. Visual similarity indexing

The visual similarity shot indexing is performed with the extraction of low-level visual descriptors capturing different aspects of human perception such as colour and texture, following the approach described in (Zhang, et al. 2008). This

approach was selected, since MPEG-7 descriptors are well established in the area of content-based search, while the employment of the R-tree structure (section 3.3.3.2) allowed for implementing retrieval in two steps (section 3.3.3.3), which was a prerequisite to apply the approach introduced in section 4.3 for combining visual and implicit feedback information. Given the fact that we followed a shot-based representation, we select the representative keyframe from each shot to extract the visual descriptors. Then, we employ an indexing structure to facilitate retrieval. In the following, we will present the visual descriptors we have extracted, the indexing structure and the retrieval functionalities.

#### **3.3.3.1. Visual descriptor extraction**

In this work, the following five MPEG-7 descriptors are generated and stored in a relational database:

- Colour Structure: it captures both the global colour features of an image and the local spatial structure of the colour.
- Colour Layout: it is a resolution invariant descriptor designed to represent the spatial distribution of colour in the YCbCr colour space.
- Scalable Colour: it is a Haar-transform based transformation applied across values of a colour histogram that measures colour distribution.
- Homogeneous Texture: it provides a quantitative characterisation of texture and is an easy to compute and robust descriptor.
- Edge Histogram: it captures the spatial distribution of edges and represents local-edge distribution in the image.

An empirical evaluation of the system's performance using different combinations of the aforementioned descriptors advocated the choice of one MPEG-7 based scheme, which relies on colour and texture and specifically the ColourLayout and EdgeHistogram descriptors are concatenated (Zhang, et al. 2008). By concatenating these descriptors, a feature vector is formulated to compactly represent each keyframe in the multidimensional space. In the following, we provide a more detailed description of the two aforementioned descriptors.

The extraction of Colour Layout descriptor (Kasutani and Yamada 2001) consists of the following stages: image partitioning, dominant colour selection, discrete cosine transform (DCT), and non-linear quantisation of DCT coefficients. In the first stage, an input image is partitioned into 64 blocks. Then, a single dominant colour is selected in each block. Subsequently, each of the three components (Y, Cb and Cr) is transformed by 8x8 DCT, and three sets of DCT coefficients are obtained. Finally, a few low frequency coefficients are extracted and quantised to form the colour layout descriptor.

Edge histogram descriptor (Eom and Choe 2005) is a histogram, where each bin corresponds to the frequency of occurrence of each of the five pre-defined edge categories in a specific region of input image. The pre-defined edge categories are: vertical, horizontal, 45o diagonal, 135o diagonal, and non-directional. First, the given image is divided into sub-images, and local edge histograms for each of these sub-images are computed. Each local histogram has five bins corresponding to the above edge categories producing a total histogram of 80 bins. To compute the edge histograms, each of the 16 sub-images is further subdivided into image blocks. A simple edge detector is then applied to each of the macro-block, treating the macro-block as a pixel. The pixel intensities for the partitions of the image block are computed by averaging the intensity values of the corresponding pixels. The edge-detector operators include four directional selective detectors and one isotropic operator. The image blocks the edge strengths of which exceed a certain minimum threshold, are used for computing the histogram.

#### **3.3.3.2. Indexing of multidimensional vectors**

Multidimensional indexing structures have been widely used for performing fast search in large scale datasets. These structures can be classified in two categories (Nam and Sussman 2004). The first includes the so-called space partitioning methods, which are based on kd-trees (Bentley 1975) and have been shown to perform well for point data. These methods aim at automatically generating an optimal partitioning of the entire multidimensional space yielding mutually disjoint sub-partitions. The second category includes the data partitioning

methods, which are based on R-trees (Gutmann 2004) and have been shown to perform well for hyper-rectangular data.

An R-tree is a height-balanced tree with index records in its leaf nodes (containing pointers to data objects). Typically, R-trees index spatial objects using their Bounding Boxes (BBs). When a query is submitted, the R-tree returns all records with BBs enclosing the query. In our case, since each keyframe (image) is represented by a  $d$ -dimensional feature vector, an R-tree structure can be constructed by associating a hyper-BB with each original image in the database. Selecting optimal hyper-BBs is crucial for the performance of the proposed retrieval system. Indeed, if the hyper-BBs are too large many of them overlap resulting in the retrieval of a large number of candidate originals and rendering the subsequent application of linear discriminant techniques ineffective. On the other hand, if the hyper-BBs are too small a similar image/keyframe is likely to fall outside the hyper-BB of its original image and not be included in the response.

An inherent drawback of R-tree based methods is the so-called dimensionality curse, which states that the computational gains in retrieval performance degrades exponentially as a function of dimensionality. For this purpose, we reduce the dimensionality of the original feature space by projecting the initial feature vectors on a fixed PCA (Principal Component Analysis) basis. We precalculate this basis by finding the principal components of the data space formed by the feature vectors corresponding to the total amount of database images and their training replicas. Given the large amount of samples, PCA manages to robustly detect the existing patterns in data and reduce the dimensionality of the indexed feature vectors without losing much of the significant information.

In this case, the PCA algorithm is applied to the involved dataset and results in a dimensionality reduction matrix  $W_d$  where  $d$  is the dimension of the new reduced vector  $f_i$ . Then the reduced features are given by  $f_i = W_d \cdot f$ . Based on the experimental work of (Nikolopoulos, et al. 2010) we reduce the feature space dimensionality to 24 and 18 dimensions for the EdgeHistogram and ColorLayout descriptors respectively. Concerning the R-tree branching factor, we have used

$M = 8$  and  $m = 4$ , as the maximum and minimum number of allowed entries (i.e., children) in a node.

### 3.3.3.3. Ranking and retrieval

The distance calculation between the descriptors of two shots is performed as described in (Mezaris, et al. 2005) by employing the functions proposed in the MPEG eXperimentation Model (MPEG-7 XM software n.d.).

Formally, when a query by visual example is initiated, the feature vector of the query shot (i.e. from the representative keyframe of it)  $Q$  is extracted and it is submitted to the R-tree indexing structure. The latter returns a set of not ranked  $K$  results  $R_V$ , which contains the shots  $s_i$ , ( $0 \leq i \leq K$ ) that are found to resemble the query one. The final visual ranking is performed by calculating the visual distances  $d_V$  between  $Q$  and all shots in  $R_V$ , so we have  $d_V(Q, i) = f_V(Q, s_i)$ , where  $f_V$  is the visual distance computing function and  $s_i \in R_V$ . The ranking for a query  $Q$  can be described as a new ordered set  $RK_V = \{s_a, s_b, s_c, \dots\}$ , where  $d_V(Q, a) \leq d_V(Q, b) \leq d_V(Q, c), \dots$  and the cardinality of  $RK_V$  elements is  $K$ .

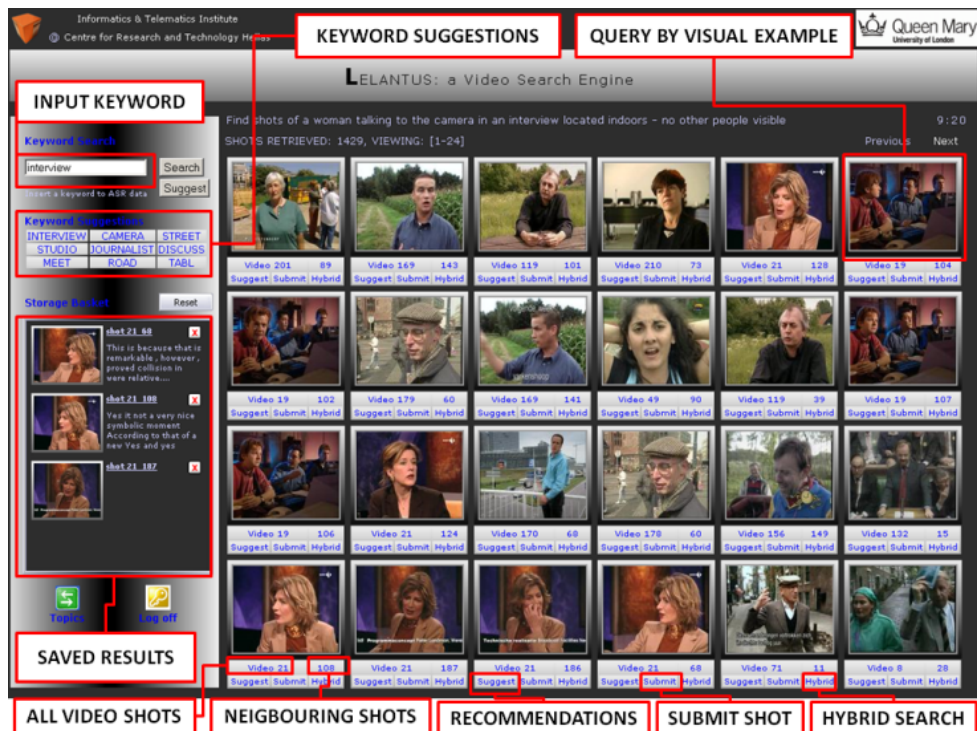


Figure 3.5. Search engine interface

### 3.4. LELANTUS interactive video search engine

In this section we present LELANTUS<sup>6</sup> interactive video search engine<sup>7</sup>, which has been implemented to realise the video retrieval experiments. LELANTUS supports searching in a video collection using the afore-described indexing techniques. The search engine is a web based video search engine and it builds upon open source technologies, while the framework and the implementation are inspired by (Vrochidis 2010c). Following the research trend in interactive video retrieval, the implemented video search engine adopts a shot-based representation. In the following, we demonstrate the interface, we discuss the retrieval functionalities and finally we provide implementation insights.

#### 3.4.1. Interface

The search engine interface through a web browser is illustrated in Figure 3.5. All the highlighted functionalities can be recorded during the user interaction. Taking a closer look we observe that the graphical user interface (GUI) is composed of two parts: the left column, which offers text-based search options and the main container, where the results are presented offering at the same time options for query submission.

At the top of the left column the user is allowed to enter a keyword in order to fire a text-based search. Two different options are provided:

- i. To perform a textual search exploiting the ASR information by pressing the “Search” button.
- ii. To search based on the semantic weights generated with the exploitation of user clicks (weighted graph, which is constructed during

---

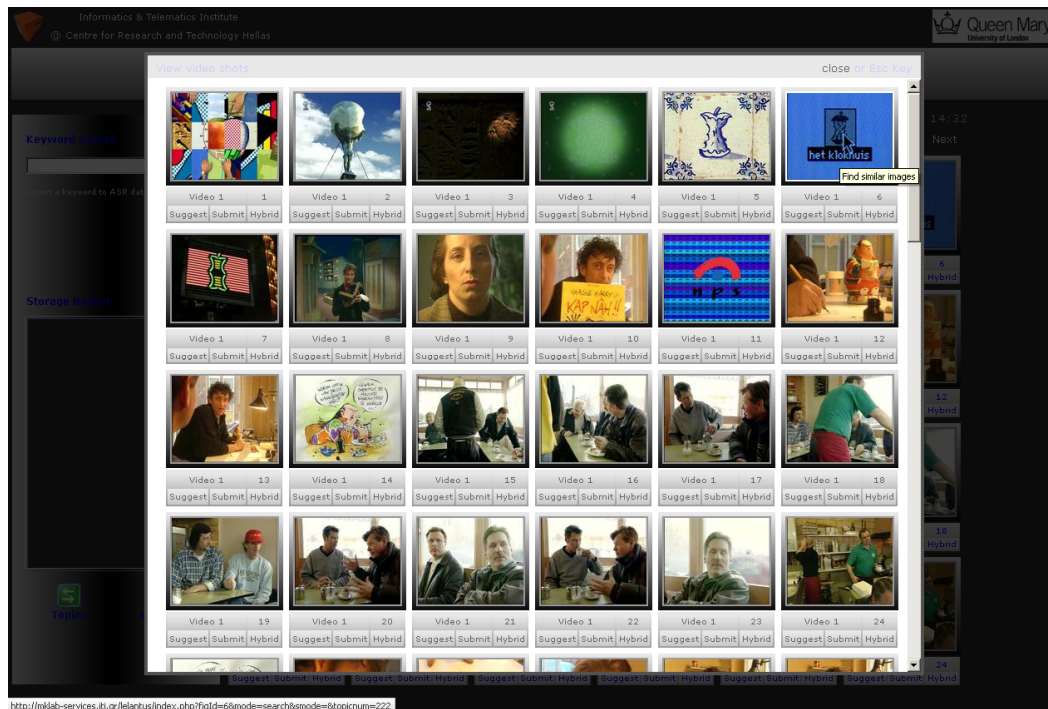
<sup>6</sup> LELANTUS in Greek Mythology was a Titan who had the capability of “moving without being seen” implying that the engine collects and processes the implicit user feedback in a transparent way to the user.

<sup>7</sup> An interactive demonstration of LELANTUS video search engine is available at: <http://mklab-services.iti.gr/lelantis>



the processing of aggregated user navigation patterns as described in section 4.2.3) and receive recommendations using the “Suggest” button.

Below this part, a number of related keywords is presented for every query submission using the weighted graph. Finally, the left column includes a basket storage structure, where the user can store the results she/he found. This basket mimics the structure that is usually available in on line stores and includes a small preview of the representative keyframe of the shot along with the associated ASR transcription.



**Figure 3.6. Keyframe-based video representation**

The main container is the part, where the results are presented. The shots are represented in an orthogonal grid by the representative keyframe. The following six different options exist for each shot and are made available to the user depending on the specific experiments:

- i. To perform a query by visual example using the analysis described in section 3.3.3 by clicking on the representative image.
- ii. To mark a shot as relevant to the topic (i.e. submit a shot).
- iii. To view all the shots of the same video on an overlaid screen as shown in Figure 3.6.

- iv. To fire a query by visual example search using the relations of the user interaction weighted graph.
- v. To execute a hybrid search, which combines visual features and implicit user feedback, as discussed in section 4.3.
- vi. To view the temporally adjacent (i.e. neighbouring) shots of a selected video shot with the associated textual transcription. This is performed thickbox component, which overlays the previous results as it is shown in Figure 3.7.

Finally, on the top part of the main container, the interface includes information on the cardinality of the results, options to navigate to next and previous pages, as well as the time that is elapsed during search.

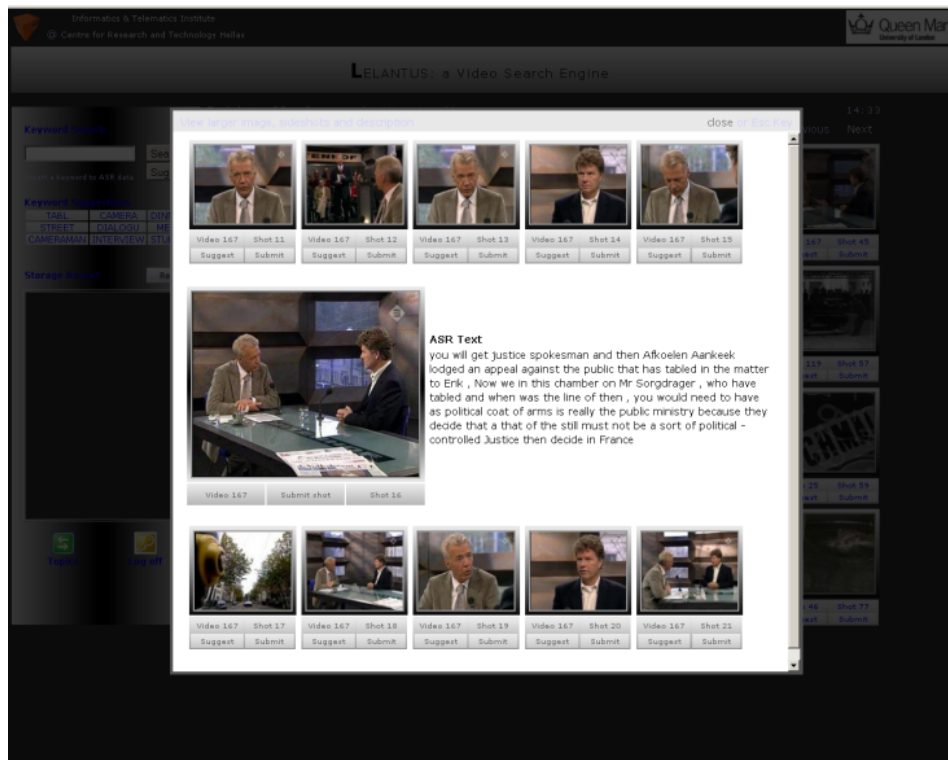


Figure 3.7. LELANTUS interface showing temporally adjacent shots

### 3.4.2. Implementation insights

LELANTUS is a web-based video search engine, which is built upon open source technologies. We have opted for a web-based implementation, since this would make the usage of the search engine more convenient following the

server-client model, while this would also facilitate its potential application as a web search engine for content available in the World Wide Web.

The following open source technologies are employed in LELANTUS: PHP<sup>8</sup>, javascript, HTML, MySQL<sup>9</sup> and Apache HTTP<sup>10</sup> server. The Apache server is deploying the application, while the source code is written in PHP, javascript and HTML. MySQL database serves as backend storage. Finally, in order to separate application logic and content we have made use of the template engine Smarty<sup>11</sup>. In other words, this template engine allowed for a modular implementation of the search system by supporting interface generation using HTML-based templates and functionality definition in PHP-based source code files.

### 3.5. Conclusions

In this chapter, we have presented the video dataset we employ to test and evaluate the algorithms that exploit implicit user feedback during interactive video retrieval. We have described the video processing and analysis techniques we applied to this dataset including shot segmentation and keyframe extraction, as well as content-based indexing based on MPEG-7 descriptors and R-trees. The aforementioned techniques have been integrated into the interactive video search LELANTUS, which is implemented to support the experiments described in the upcoming chapters. Specifically, LELANTUS is used in the interactive retrieval experiments, in which the users perform video search tasks, while their navigation patterns are recorded into log files and their gaze movements are captured with the aid of an eye-tracker. In the following chapters we present in detail these experiments and the techniques applied for the exploitation of the implicit user feedback.

---

<sup>8</sup> <http://www.php.net/>

<sup>9</sup> <http://www.mysql.com/>

<sup>10</sup> <http://httpd.apache.org/>

<sup>11</sup> <http://www.smarty.net/>

## **Chapter 4**

### **EXPLOITATION OF USER PAST NAVIGATION PATTERNS TO ENHANCE INTERACTIVE VIDEO RETRIEVAL**

*This chapter describes an approach to exploit the implicit user feedback gathered during interactive video retrieval tasks and expressed as past user navigation patterns. We propose a framework, where the video is first indexed according to temporal, textual and visual features and then implicit user feedback analysis is realised using a graph-based methodology. The generated graph encodes the semantic relations between video segments based on past user-interaction and is subsequently used to generate recommendations. Moreover, we combine the visual features and implicit feedback information by training a support vector machine classifier with examples generated from the aforementioned graph, in order to optimise the query by visual example search. The proposed framework is evaluated by conducting real user experiments. The results demonstrate that significant improvement in terms of precision and recall is reported after the exploitation of implicit user feedback, while an improved ranking is presented in most of the evaluated queries by visual example.*

#### **4.1. Introduction**

As already discussed in Chapter 2, recent works in multimedia retrieval take into account the implicit user feedback with a view to facilitating search tasks and bridge the semantic gap. In this chapter, we consider as implicit user feedback any action or navigation behaviour of the user during interactive video retrieval tasks, including mouse movements and clicks, as well as keyboard inputs and keystrokes. In this context, we propose a video retrieval framework, which

combines video analysis, as well as implicit user feedback recording and processing. Then, we provide recommendations based on past user interaction and we offer a hybrid visual search modality by combining heterogeneously extracted information (i.e. implicit feedback and visual features) by employing machine learning methods.

Video processing has already been discussed in Chapter 3. On top of this we attempt to exploit the implicit user feedback with a view to initiating semantic relations between the video segments. This is performed by introducing implicit interest indicators for video search and then by constructing a semantic affinity graph inspired by the approach proposed in (Hopfgartner, et al. 2008). Then this graph is utilised to generate recommendations in the following two steps. First, an action graph that describes the user navigation pattern is generated by employing a novel methodology that defines search subsessions (i.e. parts of sessions, in which the user searches a specific topic) based on query categorisation. Then a set of action graphs is converted to a single weighted graph by aggregating the action graphs and assigning weights to the user actions that quantify the implicit interest indicators. In order to provide recommendations, we employ a distance-based algorithm to rank the graph nodes. Additionally, when a query by visual example is considered, this graph is utilised in a similar way to define positive and negative examples. The latter are merged with a set of visually similar and dissimilar examples based on visual features, in order to construct a training set, which is used to train a Support Vector Machine (SVM) classifier that reranks the results of the visual search.

This framework is realised in an interactive video search engine (section 3.4), which supports video retrieval functionalities including text, visual and temporal search. The search engine is used for the evaluation of the approach by conducting real user experiments in 3 phases: first, a baseline system that supports only video analysis retrieval options is used by the users and their actions are being recorded; then, different users are searching for topics that are slightly different than the aforementioned ones using both the baseline and the enhanced version of the search engine, which exploits also user implicit feedback, in order to evaluate the recommendations; finally, in the third phase, another

group of users is recruited to evaluate the reranking of the visual results. Therefore, the evaluation of the proposed approach was based on the direct comparison of the baseline with the enhanced system. Additional comparisons with other relevant works have not been attempted given the fact that the latter could not be directly applied to the proposed retrieval scenario, since either they didn't consider visual search (Hopfgartner, et al. 2008), or they didn't take into account sequence-based analysis of past user interaction (Yang, et al. 2007).

The research novel contributions of this chapter are summarised in the proposed methodology of past user interaction analysis with a graph representation based on query categorisation and the definition of subsessions, as well as in the methodology for combining visual features with implicit user feedback. To the best of our knowledge this is one of the first attempts to combine patterns of past user interaction with visual features. Another relevant work that focused on combining visual features with past user interaction was performed by Urban, et al. (2006), who followed an adaptive retrieval approach to understand the user needs during the retrieval phase based on a query learning strategy. However this work did not consider aggregated user information, which differentiates our approach. In another work, (Yang, et al. 2007) presented a video recommendation system based on multimodal fusion of different sources (textual, visual and click-through data), which however does not consider sequence-based representation of aggregated past user interaction.

Parts of this chapter have been published in (Vrochidis, et al. 2010a), (Vrochidis, et al. 2010b) and (Vrochidis, et al. 2011).

This chapter is structured as follows: section 4.2 describes the processing of user implicit actions based on a graph approach and section 4.3 presents the methodology for combining visual features with graph structured implicit user feedback. The experimental results and the evaluation are presented in section 4.4, while visual results through user interaction modes with the search engine are presented in section 4.5. Finally, section 4.6 concludes the chapter.

## 4.2. Implicit feedback analysis

### 4.2.1. Implicit interest indicators

The first step towards understanding and measuring the implicit user feedback is the introduction of implicit interest indicators (Claypool, et al. 2001). The implicit interest indicators measure aspects of the user navigation patterns during a retrieval task, in order to exploit the information content that the latter carries about the user's perception of the presented multimedia material. To do that we need to identify the behaviours and the actions of a user that could declare interest and could convey meaningful information about his preferences. Based on available video search techniques (reported in section 2.1.2.1 and enhanced by temporal queries introduced in section 3.4), we define the following minimum set of user actions that can be considered as the main implicit interest indicators for video retrieval and are supported by LELANTUS:

1. Text-based query (TQ): the user inserts a keyword and submits the query. We assume that when a user submits a keyword as a search term, this keyword satisfies his/her query (or at least part of it) with a very high probability.
2. Visual query (VQ): the user selects a shot and submits a visual query by example. We assume that when a user selects a keyframe and searches for visually similar images, then she/he is also interested in the example that uses with a high probability.
3. Side-shot query (SQ): the user selects a shot in order to view the temporally adjacent shots and the associated textual description. In that case the user is very likely to be interested in the shot she/he selected.
4. Video-shot query (VSQ): the user selects a shot and retrieves all the shots of the same video. In this case we consider that she/he is interested in the initial shot to a certain extend.
5. Submit a shot (SS): the user marks a shot as relevant. In this case we assume that the user is very interested in this shot.

From the user point of view, all these actions may imply different functionalities, however they can be translated as declaration of interest on a specific shot with a

higher or lower probability. In addition to these indicators, we could also consider query-by-concept and explicit relevance feedback. However, as discussed in section 2.1.2.1, the query-by-concept actually is an extension to both query-by-textual-keyword and query-by-visual-example (since it includes textual input and uses visual features for performing retrieval), while explicit relevance feedback selections can be considered equivalent to SS indicator.

In order to interpret meaningfully each of these actions, we need to rank and quantify the levels of interest of the user to the multimedia material (shot/keyword) by associating a weight to each of these actions. In order to assign the representative weights we asked ten users to rate the level of interest for each search action in the range between 0 and 10. This level of interest actually represents the importance of each selected shot or submitted query with respect to the search topic. The users were postgraduate students and researchers (6 male and 4 female) with an average age of 29.2 and with a computer science background. All these users have been involved in TRECVID experiments in the past using similar interactive video search engines and therefore we assume that there are experienced enough to judge the importance of each functionality. The results (i.e. the average weights for each action) of this survey are presented in Table 4.1 and form the basis for the construction of the weighted graph (section 4.2.3). Although the number of users that were employed was limited due to time constraints, the results already provide an important indication of the quantification of the importance for each action.

**Table 4.1. Assign weights for each action**

Actions(a)	g(a)
Text-concept query (TQ)	7.9
Visual query (VQ)	8
Side-shot query (SQ)	7.1
Video-shot query (VSQ)	5.8

#### 4.2.2. Action graph

We exploit past user interaction information by employing an extended variation of the graph construction methodology proposed in (Hopfgartner, et al. 2008).

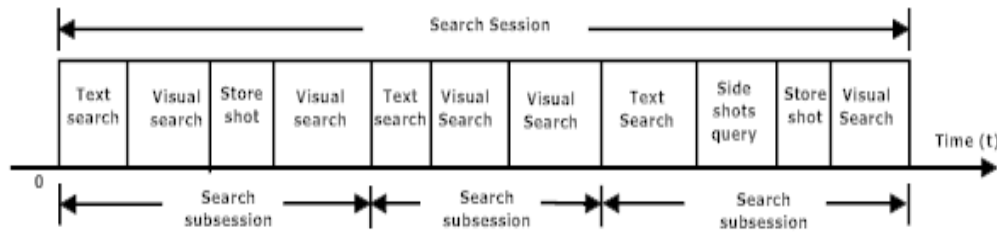


Specifically, we enhance this approach by considering query categorisation and splitting search sessions into subsessions. Although the subsession-based modelling requires an additional computational step (i.e. to identify the subsessions) compared to (Hopfgartner, et al. 2008), it reduces the complexity of the constructed graph, since not many links between the graphs representing each subsession are eventually established.

In order to describe better the proposed methodology we introduce some basic definitions. We define as “search session” the time period that a certain user spends on searching. We consider as “search subsession” the time periods a certain user spends searching for a specific topic. In addition, we provide a categorisation schema for the user actions during interactive video search tasks. First, we introduce the property of “transitivity”, which characterises an action based on its output. More specifically we consider an action as “transitive”, when it generates an output and so it satisfies the triplet:

$$input \rightarrow action \rightarrow output \quad (4.1)$$

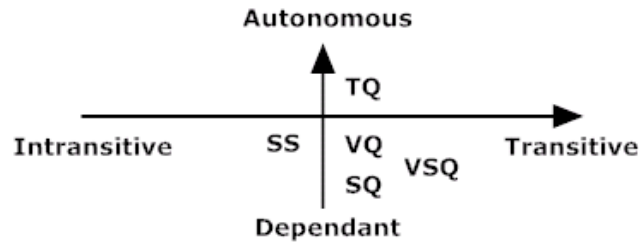
On the other hand, when an action does not provide any output, (i.e.  $input \rightarrow action$ ) it is characterised as “intransitive”. Furthermore, we classify the query actions into two main categories based on their dependency with previous actions: a) the autonomous queries, which do not depend on previous results and b) the dependent queries, which take as input results from previous search actions.



**Figure 4.1. Search session and subsessions**

To construct an action graph based on the user search activity, we exploit the properties of the involved user actions. During a search session it is possible to have a series of transitive actions, where part of the output of one action is the input for another (e.g. a result from a text search is the input for a visual search).

Consequently, to create a link between two nodes of an action graph, we need to have a sequence of two actions, where at least the first one has to be transitive. During a search session, the user may search for a specific topic, however it is possible that the user fires a search having a very broad or complex topic in mind, or even decides to change the search topic during the session. For this reason, we propose that such sessions should not be analysed as whole, but should be first decomposed into subsessions. Assuming that every autonomous query could initiate a different topic search, we propose a novel methodology, based on which, we divide each search session into “search subsessions” generating in that way several subgraphs and using as break points the autonomous queries.

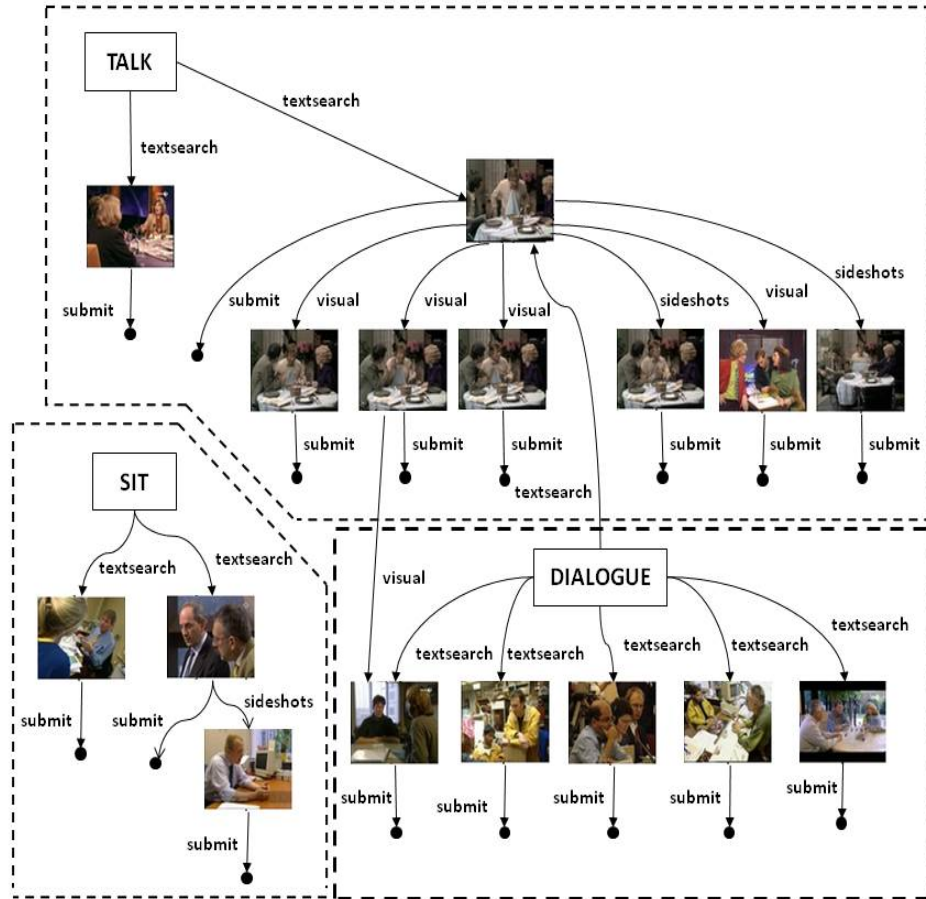


**Figure 4.2. Classification of user actions**

Taking into account the corresponding functionalities of the introduced implicit interest indicators, only the text-based search can be denoted as autonomous query, while the other queries are considered as dependent. This is because the submission of a text-based query (i.e. a set of keywords) does not necessarily depend on results retrieved by a previous query, while the other queries (e.g. visual search) depend on previous results, given the fact that the user selects a keyframe as query image from these results. In such a case, the text-based query is utilised as a break point between the subsessions as illustrated in the example of Figure 4.1. The overall classification of these functionalities can be visualised in the two different axes of transitivity and dependency as shown in Figure 4.2.

In the general case, a search subsession  $S$  consists of a set of actions  $A_S$  that includes one autonomous and a number of dependent query actions. The proposed subgraph  $G_S$  is comprised by a set of nodes (i.e. shots and keywords that represent inputs and outputs of a set of actions  $A_S$ ) and links that represent the corresponding actions  $a_i \in A_S$ , where  $i \in \{1, \dots, N_S\}$  and  $N_S$  is the cardinality of

the elements of  $A_s$ . The action graph of a search session is composed of several subgraphs, which reflect the respective subsessions and have as parent nodes the autonomous queries.



**Figure 4.3. Action graph after user interaction**

These are illustrated in the example of Figure 4.3, where an action graph for a search session is presented. Here, the user is searching for shots, in which people sitting at a table talking are depicted.

Then we construct a single action graph aggregating the action graphs from the different user sessions. More specifically, all the nodes from the individual action graphs are mapped to single action graph, and then all the action edges are mapped onto the same graph, generating in that way multiple links between the nodes. We observe that the three keywords that have been used to start the search (i.e. talk, sit and dialogue) are considered as the parents for new subgraphs, which correspond to different subsessions. In this way, concepts with different semantic meaning are not interconnected (e.g. ‘talk’ with ‘sit’), while keywords

with similar semantic meaning (i.e. ‘talk’ and ‘dialogue’) are eventually linked due to the visual similarity between two shots in different subgraphs.

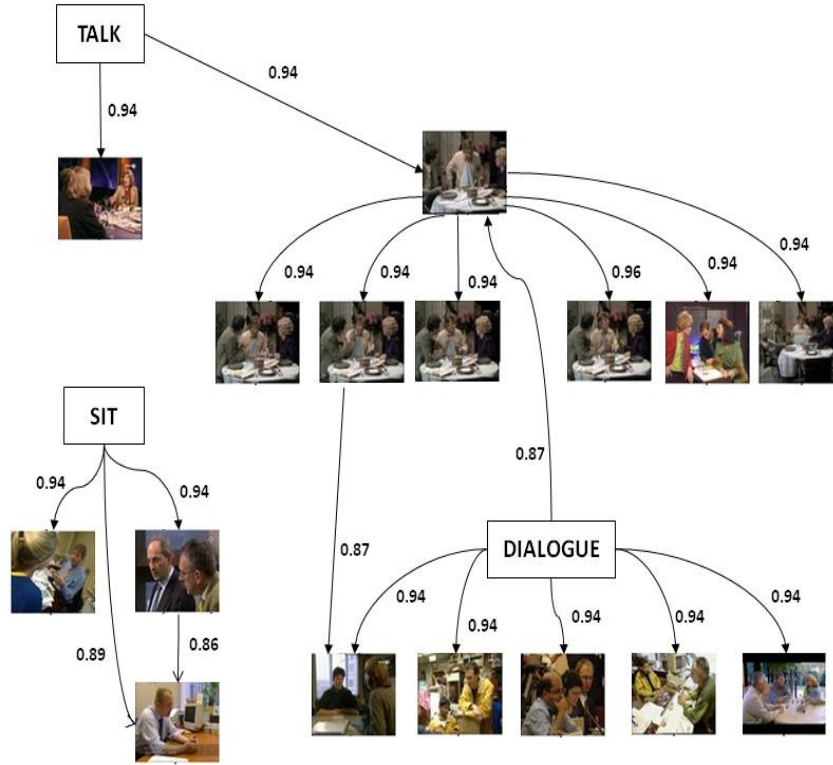


Figure 4.4. Weighted graph after processing the action graph of Figure 4.3

#### 4.2.3. Weighted graphs

After the construction of a single action graph, we generate the weighted graph in the following three steps: a) the relevant results are linked to the parent query, transforming in that way the intransitive actions into transitive, b) the multiple links between the same nodes are collapsed into one and c) actions are translated into weights.

The final weight  $w$  for a link  $n$  between two nodes  $k$  and  $m$  is given by the formula:

$$w(n) = 1 - \frac{1}{x(n)} \quad (4.2)$$

where  $x(n)$  is the sum of weights of each action that interconnects nodes  $k$  and  $m$ . This sum is expressed as:

$$x(n) = \sum_{a \in U_s} g(a) \quad (4.3)$$

where  $g$  is a function that maps each action to an implicit weight and  $a$  is an action that belongs to the set of actions  $U_s \subseteq A_s$  that comprise the different links between the nodes  $k$  and  $m$  (Hopfgartner, et al. 2008). Following the analysis of section 4.2.1, we assign indicative values (between 0 and 10) that quantify the level of interest associated to the introduced implicit interest indicators (Table 4.1). Using the equations (4. 1) and (4. 2) and the defined values for  $g$ , we are able to construct the weighted graph. Figure 4.4 illustrates the weighted graph that is produced after processing the action graph of Figure 4.3 according to the aforementioned methodology.

#### 4.2.4. Generation of recommendations

In (Vallet, et al. 2008) several recommendation algorithms based on such a weighted graph have been proposed. However, in most of the cases the authors conclude that the best performing algorithm strongly depends on the search topics. Therefore in this work, we employ a straightforward algorithm that initiates recommendations based on the distances on the weighted graph. The latter are calculated as the shortest path between two nodes. The calculation of the distances between two different nodes in this graph is performed with the application of Dijkstra algorithm (Dijkstra 1959), which computes the shorter path between two nodes. Assuming that the starting node is called the initial node, Dijkstra's algorithm assigns initial distance values and attempts to improve them step by step in the following way:

1. Assign to every node a tentative distance value. This is set to zero for the initial node and to infinity for all other nodes.
2. Mark all nodes as unvisited. Set the initial node as current. Create a set of the unvisited nodes called the unvisited set consisting of all the nodes except the initial node.

3. For the current node, consider all of its unvisited neighbours and calculate their tentative distances. Even though a neighbour has been examined, it is not marked as "visited" at this time, and it remains in the unvisited set.
4. When all the neighbours of the current node are considered, mark the current node as visited and remove it from the unvisited set. A visited node will never be checked again.
5. If the destination node has been marked as visited or if the smallest tentative distance among the nodes in the unvisited set is infinity, then stop and terminate the algorithm.
6. Select the unvisited node that is marked with the smallest tentative distance, and set it as the new "current node" then go back to step 3.

Although Floyd's algorithm (Floyd 1962) is usually faster for calculating all the shortest distances according to graph theory, it is better suited for more dense graphs. In our case the produced weighted graphs are considered to be rather sparse and can be represented more efficiently with the aid of adjacency lists instead of adjacency matrices. In that way the method scalability is also supported, as this solution should be applicable for very sparse graphs generated by a large number of users and big datasets.

Since the calculated distance is based on implicit information but it reveals semantic relations, we name it "implicit semantic distance". Hence, based on the shorter path approach, we can calculate the implicit semantic distance of each query  $Q$  that is represented as a node in the graph, with the rest of the nodes included in the graph. In this case we need to notice that the query  $Q$  can be either a shot or a keyword, while the same stands for the results. Formally, we compute  $d_I(Q, i) = f_I(Q, s_i)$ , where  $f_I$  is the implicit semantic distance computing function,  $s_i \in R_I$ ,  $R_I$  is the set of  $M$  shots or/and keywords that are interconnected through links with the query  $Q$  and  $1 \leq i \leq M$ . The ranked recommendations for query  $Q$  can be described as a new ordered set  $RK_I = \{s_{\acute{a}}, s_{\acute{b}}, s_{\acute{c}}, \dots\}$ , where  $d_I(Q, \acute{a}) \leq d_I(Q, \acute{b}) \leq d_I(Q, \acute{c}), \dots$  with a cardinality of  $M$  elements.

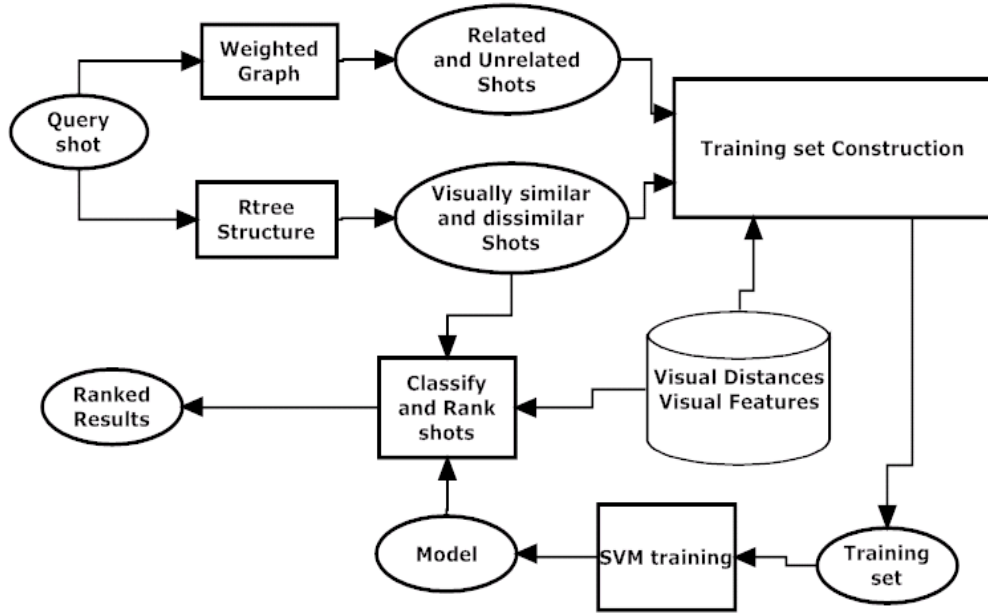
Another important functionality of the weighted graph is that it can be used to suggest new search term recommendations by calculating the distances of the input keyword term with the rest of the keywords in the weighted graph. Analogously, it can also generate related terms for a query by visual example by presenting to the user the keywords that are found to be closer in terms of distance to the query shot in the weighted graph.

### **4.3. Combining visual and implicit feedback information**

The objective of this section is to rerank the initial results of the query by visual example by exploiting the weighted graph. Although the results obtained based on visual descriptors are usually quite satisfactory, in many cases visual search fails to fetch results of the same semantic meaning confused by similar colours or textures of semantically irrelevant depictions. For instance, if we observe the visual search results illustrated in Figure 4.14, it is clear that besides the visually and semantically relevant shots (i.e. people talking on a table) to the query image at the top left corner, the output of the system also includes video shots with relevant colours, which are semantically irrelevant (e.g. road, people outdoors) to the query. Given the fact that the users expect not only visually but also semantically similar results from such queries (section 2.1.2.1) we need to optimise accordingly the visual similarity function to give more emphasis on semantically similar results.

As discussed in section 3.3.3, visual search is performed in the following two steps: i) by submitting the query descriptors to the R-tree structure and ii) by ranking the results returned calculating the distances between visual descriptors. The idea is to tune appropriately the ranking function of the second step with the aid of semantically related shots, in order to emphasise more on the specific visual features that can be of importance for each query. It is expected that training a classifier with semantically positive and negative examples from the user implicit feedback could optimise the ranking function adequately. More specifically, we train a classifier for each visual example query by employing as training set a combination of visually similar (almost duplicates) and dissimilar examples, as well as positive and negative samples generated by implicit

feedback information, in order to rerank the initial visual results. In Figure 4.5 the overall algorithm of the proposed approach is presented.



**Figure 4.5. Algorithm to combine visual features and implicit user feedback**

As shown in the diagram, when a query by shot example  $Q$  is submitted, we produce the following ranked datasets of results: two sets  $R_I$  and  $U_I$  (i.e. related and unrelated shots respectively) using the weighted graph that is constructed by processing the past user interaction and one set  $R_V$  provided by the R-tree structure that includes visually related shots according to the visual features employed. Subsequently, parts of these sets are merged as described in section 4.3.1, in order to construct a training set  $T$  and then train a support vector machine classifier utilising as features the visual descriptors. Finally, we employ the  $R_V$  (i.e. the set of results from visual search) as the test set, which is ranked according to the degrees of coefficients that are the output of the classifier and represent a similarity metric between each shot of the test set and the query. In the next subsections we provide the details about the training set construction and the SVM training.

#### 4.3.1. Training set construction

In order to train a classifier, we need to identify a proper training set  $T = T_P \cup T_N$ , where  $T_P$  is the set of the positive and  $T_N$  the set of the negative samples.



Utilizing the weighted graph, we can extract a number of positive samples that are closer to the query shot and a number of negative samples that are placed as further as possible in this graph. Hence, we create the set of positive samples  $T_{I,P} = R_{I,P}$ , where  $\forall s_i \in R_{I,P}, d_I(Q, i) < d_{I,lthres}$  and  $d_{I,lthres}$  is an experimentally set threshold. In the same way, we define a set of negative examples by employing another distance threshold  $d_{I,hthres}$  and we consider the  $T_{I,N} = U_I$ , where  $\forall s_i \in UK_I, d_I(Q, i) > d_{I,hthres}$ . In the best case, these shots should not be interconnected with the query shot in the graph (i.e.  $d_I(Q, i) = \infty$ ). Alternatively, we can select a predefined number of negative and positive samples from the graph and apply the distance thresholds only if required.

The obvious approach could be to simply train the classifier with  $T_{I,P}$  and  $T_{I,N}$ . However, due to the fact that implicit feedback is not always precise, such an approach would not always be efficient. On the other hand, visual search is usually capable of retrieving very similar keyframes (duplicates), which demonstrate an almost zero visual distance. Therefore, in order to minimise such effects and in addition to exploit the results from visual ranking that are of good quality, we follow an approach inspired by pseudo-relevance feedback technique (Xu and Croft 1996). The latter, also known as blind relevance feedback, automates the manual part of relevance feedback, so that the user gets improved retrieval performance without an extended interaction. The method is to do normal retrieval to find an initial set of most relevant documents, to then assume that the top  $k$  ranked documents are relevant, and finally to do relevance feedback as before under this assumption. Following this idea, we include in the positive samples the shots that are visually closer to the query example by employing an experimentally set threshold. Furthermore, we include in the negative samples some of the visual results that are very far from the query image taking into account again a visual distance threshold.

Formally, we construct a new set of positive samples  $T_{V,P} = R_{V,P} \subseteq R_V$ , where  $R_V$  is the set of visual search results (section 3.3.3.3),  $\forall s_i \in R_{V,P}, d(Q, i)_V < d_{V,lthres}$  and  $d_{V,lthres}$  is an experimentally set threshold. Subsequently, we define a set of negative samples  $T_{V,N} = R_{V,N} \subseteq R_V$ , where

$\forall s_i \in R_{V,N}, d_V(Q, i) > d_{V,hthres}$  . These distance thresholds could either experimentally be set to specific values, where always  $d_{V,hthres} > d_{V,lthres}$  or they could be manually adjusted by the user in a manual assisted combination of implicit information and visual data according to the user needs. The final training set is expressed as:

$$T = T_P \cup T_N = (T_{I,P} \cup T_{V,P}) \cup (T_{I,N} \cup T_{V,N}) \quad (4.4)$$

#### 4.3.2. Support vector machine classifier

Since the training and the reranking of the results are performed in real time during the query, we have to select a fast algorithm implementation, which can provide results in reasonable time. The advantage of performing the training at real time is that in a semi-automatic version of the module, the user would be able to optimise the combination procedure by adjusting weights for the two involved rankings (i.e. visual and implicit), which would reflect to the definition of different distance thresholds in the training data construction. Of course besides the implementation, the size of the training dataset comprises an important factor that could keep the speed low.

In this task we employ Support vector machines (SVMs), which have been applied with success in several classification problems (e.g. (Jiang, et al. 2010), (Ballan, et al. 2010), (Yildizer, et al. 2012)). SVMs constitute a set of supervised learning methods used for classification and regression. When a set of training positive and negative examples is available, a SVM training algorithm builds a model that predicts in which category a new example falls. To achieve this, a support vector machine constructs a hyperplane or set of hyperplanes in a high or infinite dimensional space. In general, it is assumed that the best separation is achieved by the hyperplane that has the largest distance from the nearest training datapoints of any class.

Due to the fact that we require a fast classification algorithm, which could perform training in real time, we employ the SVM implementation described in (Joachims 2006), which supports SVM training in linear time. This implementation realises the alternative structural formulation of the SVM

optimisation problem for conventional binary classification with error rate. For a given training set  $(x_1, y_1), \dots, (x_n, y_n)$  with  $x_i \in R^N$  and  $y_i \in \{-1, +1\}$ , training this binary classification SVM solves the following optimisation problem, which was proposed for predicting structured outputs and optimizing multivariate performance measures like F<sub>1</sub>-Score (section 2.1.2.4) or the Precision/Recall Break-Event Point (i.e. is the point at which precision equals recall) (Joachims, 2005).

$$\min_{w, \xi \geq 0} \frac{1}{2} w^T w + C \xi \quad (4.5)$$

$$\text{subject to: } \forall c \in \{0,1\}^n: \frac{1}{n} w^T \sum_{i=1}^n c_i y_i x_i \geq \frac{1}{n} \sum_{i=1}^n c_i - \xi \quad (4.6)$$

where  $C$  is the capacity constant and  $w$  a parameter vector. This approach has  $2^n$  constraints, one for each possible vector  $c = (c_1, \dots, c_n) \in \{0,1\}^n$  and it has only one slack variable  $\xi$  that is shared across all constraints. The algorithm that is employed to solve the aforementioned classification SVM optimisation problem is an adaptation of the Cutting-Plane Algorithm. This algorithm iteratively constructs a sufficient subset  $W$  of the set of constraints. Starting with an empty set of constraints  $W$ , in each iteration, it first computes the optimum over the current working set  $W$  (i.e.  $w = 0$  and  $\xi = 0$  in the first iteration) and then it finds the most violated constraint in the optimisation problem and adds it to the working set  $W$  (Joachims, 2006).

Assuming that the concatenated visual descriptor is represented as  $V = \{v_0, v_1, \dots, v_P\}$ , then (4.6) is transformed into:

$$\forall c \in \{0,1\}^n: \frac{1}{n} w^T \sum_{i=1}^n c_i y_i V_i \geq \frac{1}{n} \sum_{i=1}^n c_i - \xi \quad (4.7)$$

After having trained the classifier with  $T$ , we provide as test set the initial results  $R_V$  based on visual descriptors. This test set is finally ranked based on the distances that are calculated between each shot and the hyperplane, which is constructed by the model.

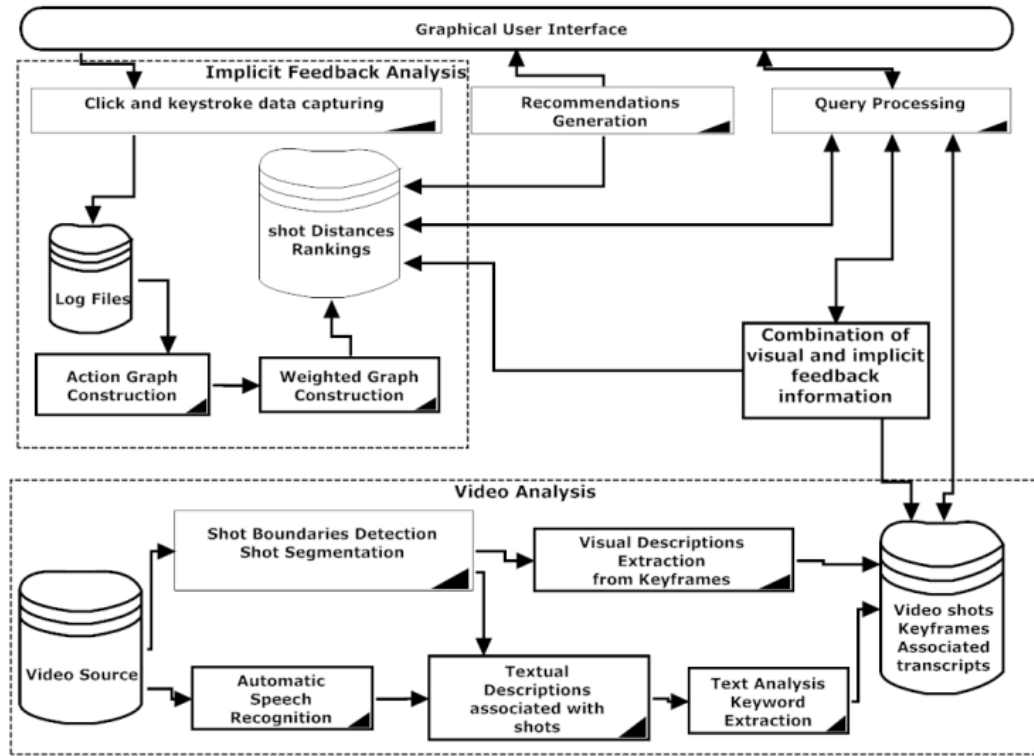


Figure 4.6. Video indexing and retrieval framework

## 4.4. Experiments and results

In this section we present the experimental framework, the results and the evaluation of the proposed techniques.

### 4.4.1. Interactive video retrieval framework

In order to evaluate the current work and perform tests and experiments we designed a video retrieval framework that supports both video content analysis and implicit feedback processing. The framework is illustrated in Figure 4.6.

As it can be observed the video analysis layer realises the processing described in Chapter 3, while the implicit feedback analysis implements the graph-oriented approach described in section 4.2. Then, we employ the LELANTUS video search engine (section 3.4) to conduct experiments and make comparisons between the different retrieval modalities. The evaluation experiment is divided into 3 phases: a) the training phase, in which the implicit user feedback is recorded, b) the evaluation of the generated recommendations and finally c) the evaluation of the hybrid search modality.

In these experiments we made use of the annotated video set of TRECVID 2008, which is described in detail in section 0.

#### 4.4.2. Training phase

In the first phase (i.e. the training phase) we have recruited 24 users (18 male, 6 female), who searched for 6 different topics. The participants were mostly postgraduate students or postgraduate researchers with an average age of 30.2 years old. All of them had a very good knowledge of English and a computer science background. In addition, most of them had a good understanding of retrieval tasks and were familiar with multimedia search engines.

**Table 4.2. User-topic assignments**

User/ Topic	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
A																								
B																								
C																								
D																								
E																								
F																								

Each user searched for 15 minutes for each topic. The assignment of the topics to the users is shown in Table 4.2. As it is illustrated, all the topics are searched 4 times by different users. Before the experiment the users have spend 15 minutes to familiarise themselves with the search engine, the retrieval functionalities and the type of the search task. Since many of these users had participated in the past in TRECVID interactive video retrieval experiments a more detailed tutorial of the search engine was not required.

During the experiment the actions of the users including mouse clicks and keyboard inputs were recorded. The users were instructed to search for following topics and retrieve as many relevant results they can:

- A. Find shots of 3 or fewer people sitting at a table.
- B. Find shots of one or more people with mostly trees and plants in the background; no road or building visible.
- C. Find shots of one or more people where a body of water can be seen.

- D. Find shots of a woman talking to the camera in an interview located indoors - no other people visible.
- E. Find shots of one or more pieces of paper, each with writing, typing, or printing it, filling more than half of the frame area.
- F. Find shots of a person on the street, talking to the camera.

In this phase, the baseline version of LELANTUS was used. This included the textual, visual and temporal retrieval functionalities as these were presented in Figure 3.5. The functionalities that are based on implicit user feedback have been disabled during this phase.

**Table 4.3. Numerical statistics for the weighted graph**

Weighted Graph	
Nodes	1298
Shots	1229
Keywords	69
Links	2659

After having recorded the navigation movements of all users, we employ the proposed methodology to generate the action and the weighted graph. The numerical statistics (e.g. number of nodes, links, etc.) of the weighted graph are reported in Table 4.3.

#### **4.4.3. Recommendations evaluation**

In the second phase (testing phase), we recruited 8 different users, who searched for 4 different topics. These participants were again postgraduate students and researchers with an average age of 29.6 years old. All of them had a computer science background, a very good knowledge of English and were familiar with search engines.

These topics were selected in such a way so that two of them were relevant (but not identical) to the ones of the first part and the two of them irrelevant. The topics of this phase are the following:

1. Find shots of one or more people with one or more horses

2. Find shots of one or more people with one or more books.
3. Find shots of a map (i.e. relevant but not identical to topic E)
4. Find shots of food and/or drinks on a table (i.e. relevant but not identical to topic A)

As it is shown, the topics 3 and 4 are considered relevant but not identical to topics E and A respectively, while the rest of the topics (i.e. 1 and 2) are considered irrelevant to all the topics A-F. In a similar way with the training phase, before the experiment, the users were familiarised with the search engine by having a tutorial session of 15 minutes to understand the retrieval functionalities and the purpose of the search tasks. The 15 minutes tutorial session was adequate, since many of these users had participated in the past in TRECVID interactive video retrieval experiments and therefore they were familiar with such tasks.

**Table 4.4. Latin square user experiment**

System/Topics	A	B	C	D
Baseline	1-4	1-4	5-8	5-8
Enhanced	5-8	5-8	1-4	1-4

The evaluation methodology followed is to compare the baseline with the enhanced version of LELANTUS. The enhanced version augments the baseline system by offering the recommendation functionality, which could retrieve results based on the weighted graph from the aggregated implicit feedback (created in the “training phase”). In this phase all the users searched 10mins for each topic. To deal with the searcher effect we have applied a latin square design as this is shown in Table 4.4. For instance, the first group of users (1-4) has searched for topics A and B using the baseline system and for topics C and D with the enhanced system. To deal with the learning effect we have assigned the users with different search topic sequences. These sequences are the same for groups 1-4 and 5-8 and have been selected in such way so each of the 4 users is starting to search with a different topic. The topic search sequences are reported in Table 4.5.

Then we compare the results provided for the two different systems. In Table 4.6 we can see the results in terms of precision, recall and F-score. In Figures 4.7 and

4.8 we show the precision and recall respectively. These metrics are calculated against the annotated results for each topic.

**Table 4.5. Topic search sequences**

Users	Topic Sequence			
1	1	2	3	4
2	4	1	2	3
3	3	4	1	2
4	2	3	4	1
5	1	2	3	4
6	4	1	2	3
7	3	4	1	2
8	2	3	4	1

**Table 4.6. Precision and recall for the baseline and enhanced systems**

Topics	Precision		Recall		F-Score	
	Baseline	Enhanced	Baseline	Enhanced	Baseline	Enhanced
1	0.956	0.896	0.042	0.045	0.08	0.086
2	0.885	0.9	0.13	0.135	0.23	0.2344
3	0.597	0.665	0.079	0.096	0.139	0.168
4	0.708	0.767	0.034	0.075	0.064	0.137

The average improvement in recall for the first two topics 1 and 2 (i.e. the irrelevant to the initial ones) is about 5%, while precision seems to slightly drop by an average of 2%. As expected, the major improvement is reported in the topics 3 and 4 (i.e. the relevant to the initial queries), in which recall and precision are increased by an average of 72% and 9.8% respectively. Concerning the F-score, this is improved in average about 4.7% for the 2 irrelevant topics and is boosted by an average of 66.7% for the relevant ones. In general, it seems that regardless of the similarity between the train and the test topics, it is evident that the consideration of past user interaction improves the system performance for new users.

It should be noted that the low absolute recall values are due to the fact that the many shots that are relevant for each query-topic, could not possibly be retrieved in the requested time duration of the experimental search sessions of this phase.



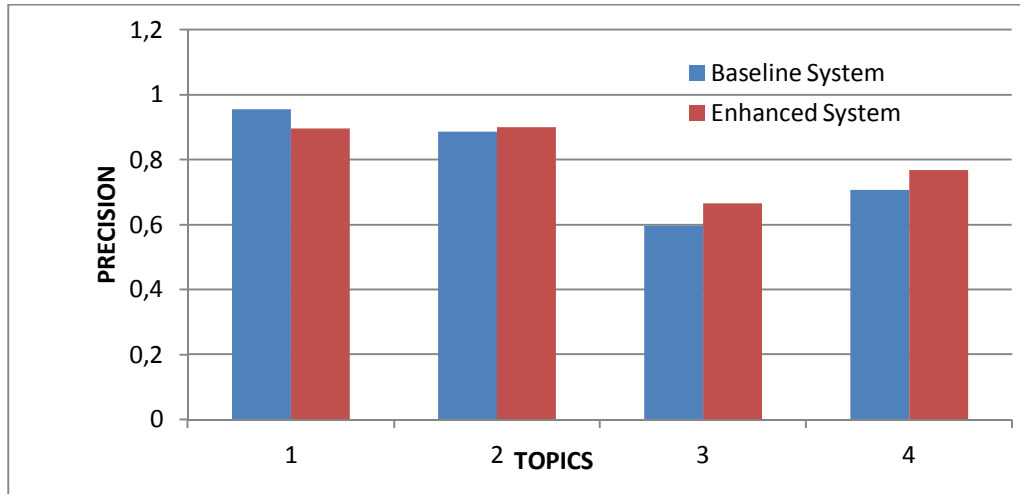


Figure 4.7. Precision for the results of the baseline and enhanced systems

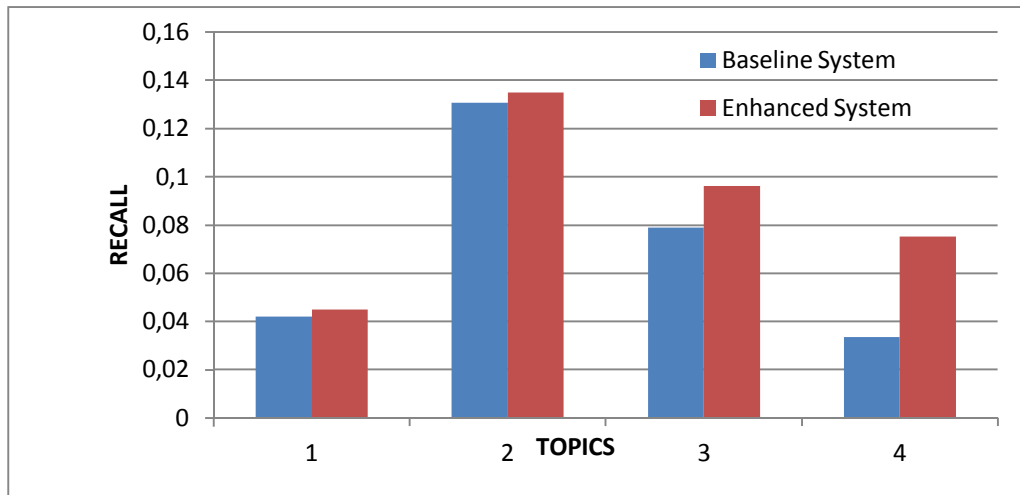


Figure 4.8. Recall for the results of the baseline and enhanced systems

#### 4.4.4. Visual search optimisation experiment

In this experimental phase we attempt to evaluate the hybrid search modality that combines visual features and implicit user feedback expressed in past user navigation patterns by comparing it with the standalone visual search functionality (section 4.3). To measure and evaluate the performance of the suggested algorithm, we compare the two different rankings in the case of a query by visual example: a) the visual ranking, which is provided by the distances of the visual indexing as this is described in section 3.3.3 and b) the hybrid ranking, which is generated after the application of the algorithm discussed in section 4.3.

To compare the aforementioned retrieval functions, we utilise the evaluation methodology suggested in (Joachims, 2002a). In this work, the authors propose to use a combined ranking that integrates two rankings that are to be compared. This form of presentation leads to a blind statistical test so that the clicks of the user demonstrate unbiased preferences. Specifically, in order to compare two rankings A and B, we combine them into a single ranking C so that the following condition holds for any top  $m$  links of the combined ranking. The top  $m$  links of the combined ranking C contain the top  $a$  links from A and the top  $b$  links from B, with  $|a - b| < 1$ . In other words, if the user scans the links of C from top to bottom, at any point she/he has seen almost equally many links from the top of A as from the top of B. It is shown in (Joachims 2002b) that such a combined ranking always exists and that it can be constructed efficiently.

Considering that  $RK_H$  is the hybrid ranking and  $RK_V$  the visual, we construct a “combined ranking”  $RK_{H,V}$  that includes the top links of both rankings. More specifically,  $RK_{H,V}$  consists of  $l$  results, which are actually the  $k_a$  top results of  $RK_H$  and the  $k_b$  of  $RK_V$ , where  $|k_a - k_b| \leq 1$ . Such method can be considered even more appropriate, when applied in visual query by example instead of text web search, since in this case, the results are not so subjective to what the user has in mind. In section 4.5 we provide indicative illustrations of a visual, a hybrid ranking and the respective combined ranking for the same query shot (Figures 4.14, 4.16 and 4.17 respectively).

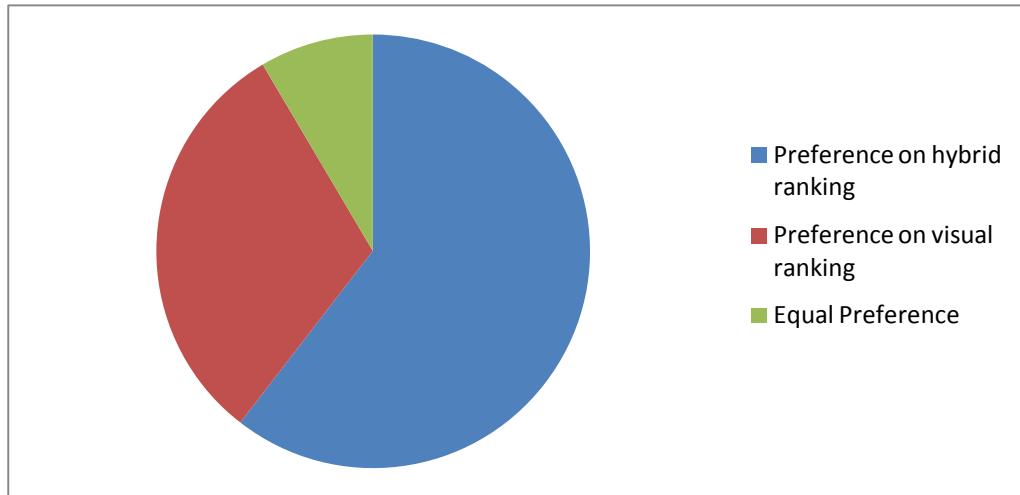
**Table 4.7. Pairwise comparison of the hybrid retrieval function with the visual one**

Total Queries	200
More selections on Hybrid	121
More selections on Visual	62
Equal selections	17

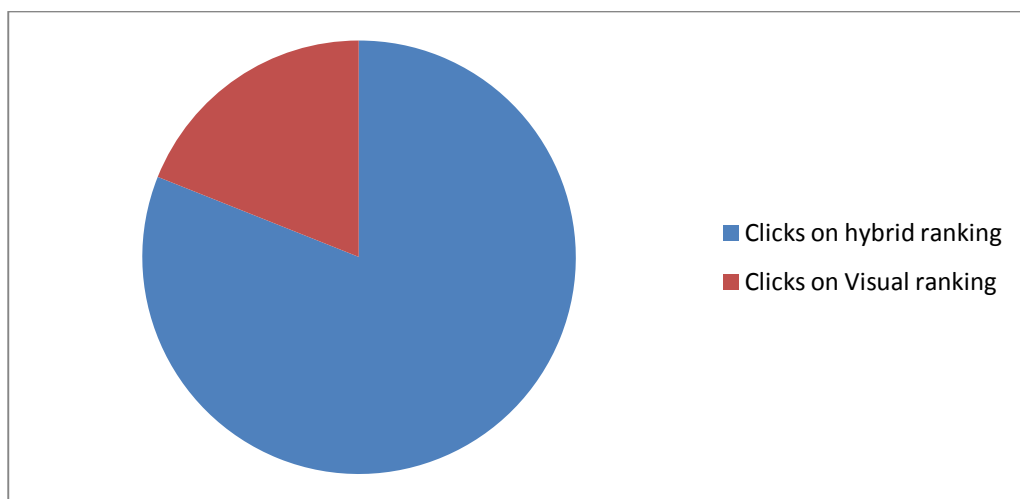
In this phase, 12 users (8 male and 4 female) were recruited. The participants were mostly postgraduate students or postgraduate researchers with an average age of 30 years old. All of them had a computer science background and they were familiar with search engines.

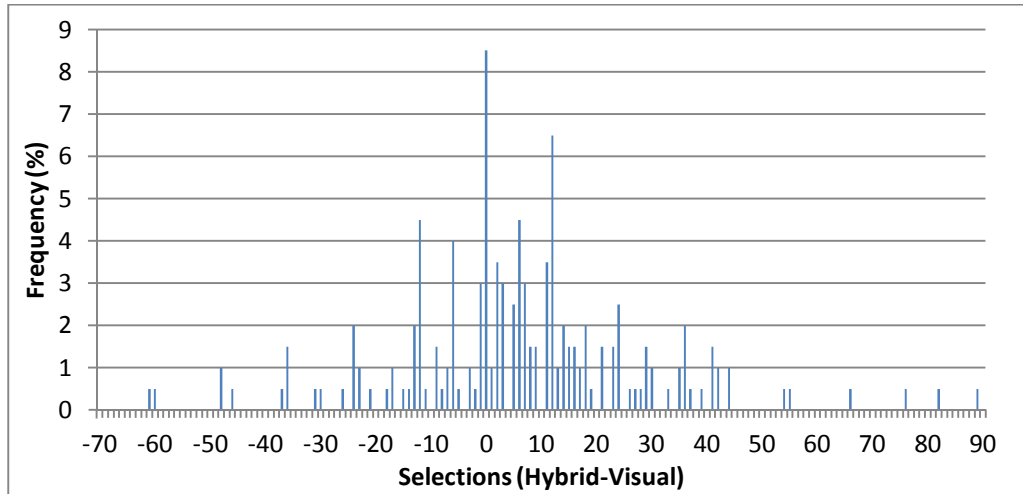
**Table 4.8. Clicks on hybrid and visual ranking**

Total Clicks	6509
Selections on Hybrid ranking	5276
Selections on Visual ranking	1233

**Figure 4.9. A visual representation of the user preference on the hybrid and visual rankings**

The users were called to identify visually similar results for 200 randomly selected queries by visual example that are included in the weighted graph. To construct efficiently the training set, the thresholds have been experimentally selected, while 30 positive and 30 negative examples are extracted by the weighted graph for each query and considered as the online training set for each query.

**Figure 4.10. A visual representation of the clicks on the hybrid and visual ranking**



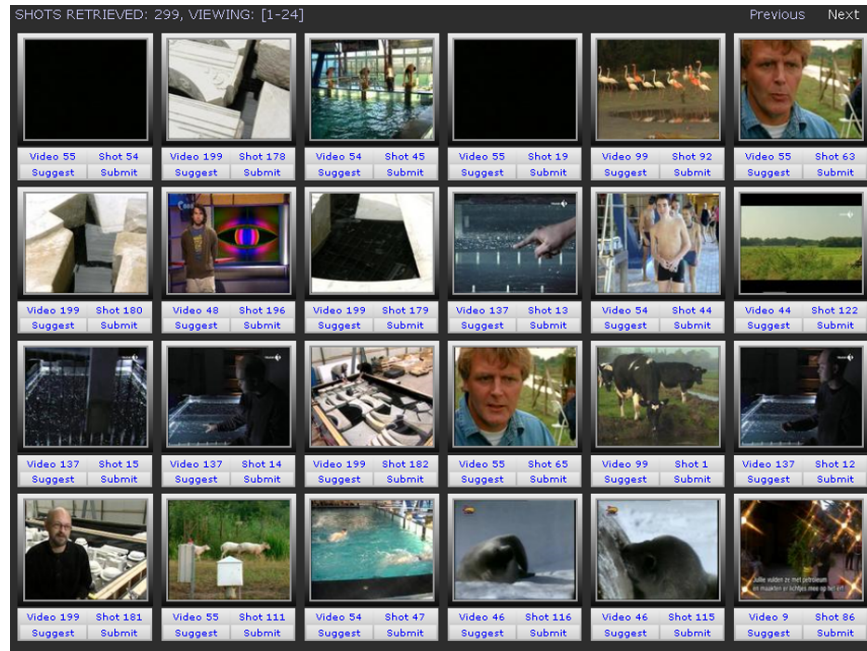
**Figure 4.11. Histogram of the clicks. The horizontal axis stands for the number of clicks (positive for hybrid ranking and negative for visual), while the vertical for the frequency**

#### **In Tables 4.7 and**

Table 4.8 we can observe the statistical results for all the users. It can be seen that for 60.5% of the queries the hybrid ranking is preferred by the users, for 8.5% of queries the two functions seem to perform equally, while for the rest 31% of queries the visual ranking outperforms the hybrid. Despite the fact that the number of the users that were in favour of visual ranking was significant, the number of the users the preferred the hybrid one was almost double. Figure 4.9 presents a visual representation of the user preferences with respect to the two rankings, while Figure 4.10 illustrates a pie of the clicks performed for the two different rankings.

In Figure 4.11, we present the corresponding histogram, which shows the frequency of user preference (i.e. selection clicks on hybrid ranking minus clicks on visual ranking) for the involved queries. We constructed the histogram by considering absolute values of the clicks and not normalised (i.e. divided by the total clicks in a query). The reason behind this choice is the fact that the actual number of clicks seems to be of more importance. For instance, in the case that a user clicks and selects only one item more from the one of the rankings, the conclusion should be that this ranking is with higher probability slightly better than the other. This can be reflected when considering the absolute difference of

the clicks (e.g.  $9-8=1$  and  $1-0=1$ ), where the value 1 describes the user preference. However, if we normalise the metrics according to the total number of selections in a query, we could get misleading results (e.g. 11,1% and 100% respectively for the previous example).



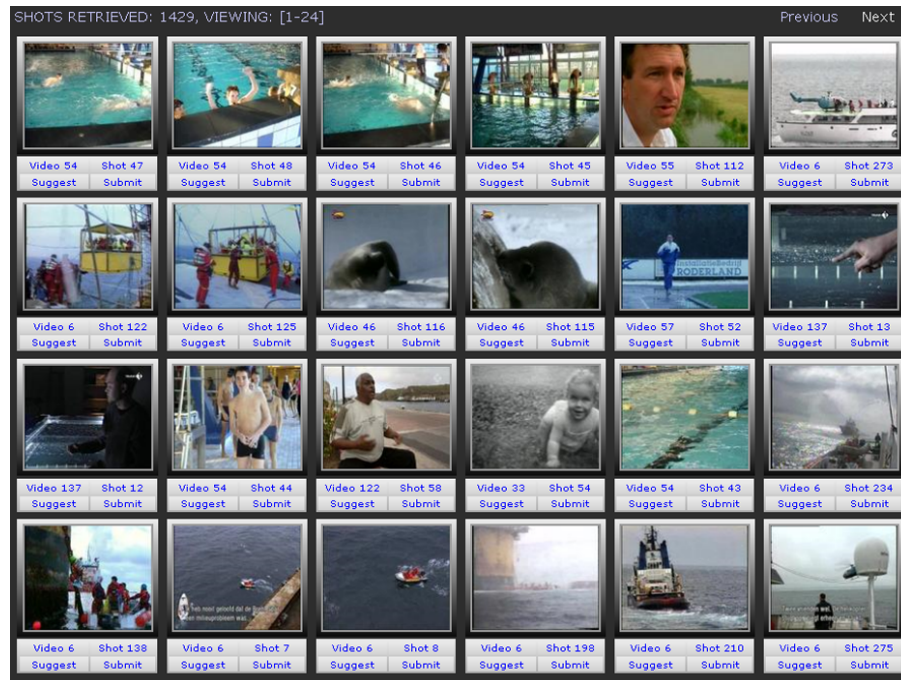
**Figure 4.12.** The user submits a textual query with the keyword “water” searching in the ASR transcripts

#### 4.5. Interaction modes

In this section, the improvement of results, when past user interaction is taken into account is demonstrated by presenting different interaction modes and by reporting results by considering the precision in the first  $N$  results ( $P@12$ ). In these examples, the system is accessing the TRECVID 2008 test data collection (presented in detail in section 0) The implicit user feedback information has been gathered during the first experimental training phase described in section 4.4.2.

First, we present a usage scenario, in which the user is searching for scenes where a water body is visible by typing the keyword “water” (Figure 4.12). As text retrieval is performed on the noisy information provided by Automatic Speech Recognition (ASR), only some of the results depict water scenes.

Conducting the same query utilising the graph with the past interaction data (i.e. the recommendations), we get a clearly better set of results (Figure 4.13).



**Figure 4.13. Results from a textual query (keyword “water”) searching with the aid of the weighted graph**



**Figure 4.14. Query by image example. Content-based analysis is employed for this query. The input image is the one on the left top corner**

At the same time the system outputs term recommendations using the weighted graph (Figure 4.18 (left)) and the keyword nodes of the graph. In this case, the



query keyword was “water” and most of the recommended words seem to have high semantic similarity with the input term (e.g. swim, ocean, beach).



**Figure 4.15. Query by image example. Relations from the weighted graph are used to realise the query. The input image is the one on the left top corner.**



**Figure 4.16. Hybrid search combining visual features and implicit user feedback. The input image is the one on the left top corner.**

In the second usage scenario, the user is employing the query by visual example methodology to find images that are similar to a given one. Subsequently, we present the set of results when the user searches with the three different modalities: a) content-based search using the visual features, b) graph-based recommendations utilizing the past user interaction and c) hybrid search, which

combines visual features and implicit user feedback. The output of the system is visualised in Figures 4.14, 4.15 and 4.16 respectively for the aforementioned search flavours, while the related terms generated by the system for this specific query are illustrated in (Figure 4.18 (right)).

If we attempt to comment on the visual output of the system, it seems that the top 12 results of the hybrid approach are of better quality compared to the results of the other search modules. More specifically a precision  $P@12$  of 91.67% (11/12) is reported for the hybrid modality, while the visual-based search achieves a lower precision of 58.3% (7/12).

Of course, the good performance of the hybrid approach is due to the high precision 75% (9/12) reported by the recommendations (Figure 4.15), as the latter are actually used for the construction of the training set described in subsection 4.3.1.



**Figure 4.17. Combined Ranking of results for the query shot on the top left corner**

In this specific case, it seems that when retrieval is performed only by considering the visual descriptors, low distances are estimated also for shots that have similar colours but no semantic resemblance with the query. On the other hand, when the hybrid ranking is applied, it seems that the implicit user feedback has given a semantic flavour to the ranking as the shots that shared only common



colour characteristics were ranked lower. As far as the term recommendations are concerned, most of the suggested words (i.e. 7 of 9) seem to be semantically related with the selected image (Figure 4.18(right)).



**Figure 4.18.** Keyword suggestions in a text query (left) and in an image by example query (right)

## 4.6. Conclusions

In this chapter we have introduced new implicit interest indicators for video search and proposed a novel methodology that considers query categorisation in order to construct a content similarity graph based on the implicit indicators of patterns of user interaction. In addition, we have proposed an approach for combining effectively visual features with implicit user feedback by employing a SVM classifier.

As it is shown by the results, the implicit user feedback expressed by aggregated user navigation patterns can be of added value in video retrieval engines, considering also that large quantities of past user interaction data can easily become available. From the experimental results it seems that the utilisation of implicit feedback is capable of improving the visual search results in most of the cases and in addition it improves the system's performance. We could say that utilising implicit user feedback to optimise visual search seems to tune the visual function in a semantic way, in which results with the same semantic concept are ranked higher despite the initial lower visual resemblance. Although the experiments were performed with a specific set of visual features (i.e. MPEG-7) it is an indication that similar performance could be expected when other features are applied, given the fact that all the low-level representations suffer from the problem of semantic gap.

## **Chapter 5**

### **INVESTIGATING AGGREGATED GAZE-BASED IMPLICIT FEEDBACK IN INTERACTIVE VIDEO RETRIEVAL**

*This chapter investigates the role of gaze movements as implicit user feedback during interactive video retrieval tasks performed in not strictly controlled environments. In this context, we use a content-based video search engine to perform an interactive video search experiment, during which, we record the user gaze movements with the aid of an eye-tracking device and generate features for each video shot based on aggregated past user eye fixation data. Then, we employ machine learning techniques in order to train a classifier that could identify shots marked as relevant to a new query topic by new users. The results of the approach are evaluated by computing the accuracy of the classifier, as well as precision and recall. The evaluation shows that important information can be extracted from aggregated gaze movements during video retrieval tasks even in not controlled environments.*

#### **5.1. Introduction**

The objective of this chapter is to investigate the potential of exploiting the implicit user feedback expressed by gaze movements during interactive video retrieval tasks. The eye movements of the users can be recorded by special devices called eye trackers. Several methods exist for recording eye movements. The most popular ones consider video images, from which the eye position is extracted, while other methods use search coils or they are based on the electrooculogram (i.e. the electric signal that is derived using two pairs of contact electrodes placed on the skin around one eye).

Inspired by the literature discussed in detail in section 2.2.2.2, the idea is to employ an eye tracking device in order to capture gaze movements of past users during interactive video retrieval tasks and subsequently extract gaze-based features and identify shots of interest in the context of specific topics. In other words, our aim is to distil meaningful information from aggregated gaze data, which could be exploited for identifying items that are of interest to a user with respect to her/his query topic. We propose an approach, in which, the gaze movements of past users are processed, in order to extract fixations (i.e. the eye remains fixed on a specific point for a certain amount of time) and pupil dilations. Then, we propose the extraction of a set of features that describes each video shot based on fixation characteristics and complemented by pupil dilation during fixations. Subsequently, we employ a Support Vector Machine (SVM) approach to train a binary classifier that could predict, which of the items viewed by a new user could be classified as interesting for her/him and matches the topic she/he searches for. The positive results of the classifier are provided as shots of interest for specific topics. To eliminate the searcher effect, we average aggregated fixation information by different users searching for the same topic. We evaluate this approach by conducting a video retrieval experiment in a not strictly controlled environment. In this experiment the users are recruited to perform video search with an interactive video search engine, while their gaze movements and pupil dilations are captured with the aid of an eye tracker.

The main contribution of this work is the methodology for processing aggregated gaze data of past users, which combines gaze fixation and pupil dilation information, in order to detect items that are relevant to a given query topic and could be utilised as recommendations for a new user. In addition, the application of eye-tracking techniques in video search experiment, which is conducted in a less controlled environment compared to other approaches (e.g. (Zhang, et al. 2010)) can be considered of importance regarding the effectiveness of the method, as well as the potential of gaze-based implicit feedback, considering that the related works in the area have investigated only strictly controlled environments so far. Parts of this work have been published in (Vrochidis 2011).

This chapter is structured as follows: section 5.2 describes the analysis of gaze movements and we introduce the SVMs employed in this approach, while section 0 presents the experiment conducted. The results and the evaluation are presented in section 5.4 and finally, section 5.5 concludes the chapter.

## **5.2. Gaze-movements analysis**

### **5.2.1. Eye movements**

Generally, the eye movements can be categorised according to the following ocular behaviours: fixations, saccades, pupil dilation, and scan paths.

Fixations are defined as a spatially stable gaze lasting at least 100 milliseconds, during which visual attention is directed to a specific area of the visual interface. Fixations are traditionally understood to be indicative of where a viewers' attention is directed, and represent instances, in which information acquisition and processing is able to occur (Rayner 1998). Based on existing literature, a very high correlation has been found between the display item being fixated and the one that is in the mind of the viewer. In addition, there is a close connection and correlation between the amount of time duration of the fixation on certain items and the degree of cognitive processing (Just and Carpenter 1980). Eye fixations are the most relevant metric for evaluating information processing primarily, since other indices, such as saccades, occur too quickly to absorb new information (Rayner 1998). At least three processes occur during an eye fixation: a) encoding of a visual stimulus, b) sampling of the peripheral field, and c) planning for the next saccade (Viviani 1990). Research has shown that information complexity, task complexity, and familiarity of visual display will influence fixation duration (Duchowski 2002). The time duration of an eye fixation is also largely dependent on a users' task. The average fixation duration during silent reading is approximately 200 ms, while other tasks, including typing, scene perception, image viewing and music reading approach averages of 300 milliseconds. From an eye-tracking perspective, multimedia information retrieval seems to encompass visual inspection on images and text, so it is expected that the average fixation duration will fall within the range of these two

groups. The differences in fixation duration can be attributed to the time required to absorb necessary information, as well as to the speed at which new information should be absorbed. While during reading it is necessary for the eye to move rapidly, in visual inspection and search, it is less imperative that the eye quickly scans the entire scene, but rather that the user can absorb key information from certain regions.

On the other hand, saccades, which are the continuous and rapid movements of eye gazes between fixation points, are believed to occur so quickly across the stable visual stimulus that only a blur would be perceived. Because saccadic eye movements are extremely rapid, within 40-50 milliseconds, and approaching velocities of nearly 500 degrees per second, information acquisition is unable to occur during this time. This lapse of information intake is traditionally referred to as “saccadic suppression”, however, due to the fact that saccades represent such short time intervals, individuals are unaware of these breaks in information perception (Rayner 1998).

Pupil dilation is a measure that is typically used to indicate an individual’s arousal or interest in the viewed content matter, with a larger diameter reflecting greater arousal (Duchowski 2002), (Rayner 1998), (Hess and Polt 1964). Studies usually compare the average pupil dilation that occurs in a specific area of interest with the average pupil dilation of the entire site to gain insight into how users might cognitively understand or process the various content matter (Hess and Polt 1960).

Finally, scanpath encompasses the entire sequence of fixations and saccades, which define and represent the pattern of eye movement across the visual scene. The behaviour of user scanpath provides insights into how a user navigates through the visual content. Studies analysing properties specific to scanpath movement have enabled researchers to create a more comprehensive understanding of the entire behavioural processes during a visual search or scanning session (Josephson and Holmes 2002). Existing literature suggests that scanpath movement is not random, but is highly related to a viewer’s frame of mind, expectations, and purpose (Yarbus 1967). In the case that the user is looking at particular content areas, several studies exploring eye movement

locations determined that unique regions of a visual item are stably viewed (i.e. a fixation is identified) sooner than others (Antes 1974).

Based on the aforementioned literature and discussion it seems that fixations and pupil dilation comprise the most reliable indicators of user interest during information retrieval tasks. In the following, we will observe and analyse how fixations and pupil dilations occur during interactive video retrieval tasks.

### **5.2.2. Fixation analysis in video retrieval**

Based on the discussion of the previous section, a very high correlation has been identified between the display item being fixated and the one that is in the mind of the viewer. In the case of interactive video retrieval, we can assume that the user focuses his/her gaze on the items that are of interest with respect to what she/he searches for. During a video retrieval session the user interacts with the visual interface of a search engine. As discussed in section 2.1.2.3 most of the video retrieval interfaces adopt a shot-based representation and therefore the videos are represented with the aid of keyframes.

After visualising and inspecting several fixation patterns on a video retrieval interface by different users, we come to the conclusion that many parts of the graphical interface are viewed constantly for a specific amount of time (i.e. a fixation point was identified). However, it is apparent that not all of them could be considered as items of interest. For instance, the user might be distracted by shots that are of interest for him but not in the context of the specific query, or she/he might steadily look parts of the interface that support query submission or page change. In addition, the user might be distracted by external factors or thoughts that cross his/her mind make him absent-minded and cause unpredictable behaviour for several seconds. This inspection took place on the gaze movements of the users employed in an eye-tracking experiment (section 0) and performed retrieval using the LELANTUS video search engine, which supports a shot-based interface.

This is more clearly shown in the example of Figure 5.1, in which a user is searching for video scenes that depict books. After the analysis of the gaze movements of a certain user, many fixations are identified, pointing at different

parts of the interface. It is obvious that although many fixations on relevant items are reported (i.e. the two shots depicting people showing or reading books on the top left corner of the interface), it is also clear that some of the video shots that draw the attention of the user (as shown by the fixations) are not relevant to the query. For instance, fixations are also identified on the shots on the top right corner of the interface, which do not illustrate books, as well as on parts of the interface that support query submission (left column). Although we are able to simply discard the fixations that do not correspond to the video shot presentation grid, we still have to face the problem of having fixations identified on non relevant shots.



**Figure 5.1. The user is searching for video scenes depicting books. The fixations are presented as blue spots on the interface**

This can be considered as a classification problem, in which we need to discriminate between relevant and irrelevant items to a query topic. Based on previous studies (Klami, et al. 2008), (Zhang, et al. 2010), eye fixation-based features have shown discrimination power over items of interest for a user in controlled image retrieval environments. Therefore, a reasonable approach is to

extract fixation-based features for each shot. In order to overcome the problem of noise introduced by the eye-tracker, the different duration of fixations for different users (due to the time required to absorb necessary information), the searcher effect and the unpredicted behaviour of the user, we propose to aggregate gaze information from multiple users searching for the same topic.

### **5.2.3. Pupil dilation analysis**

As already discussed in section 5.2.1, fixations do not comprise the only ocular behaviour that reveals user interest. Based on the literature, cognitive behaviour can also be inferred by the pupil dilation and to a less extent by saccades and scan paths. Therefore and in order to complement the fixation information we propose to take also into account the pupil dilation of the user.

As already discussed, many research studies have documented that emotional and sensory events elicit a pupillary reflex dilation (Krenz, et al. 1985), (Loewenfeld and Lowenstein 1993), (Smith, et al. 1970). More specifically, recent experiments in (Privitera, et al. 2008) showed that a significant pupil response is reported for visual target detection events. This means that a strong correlation can be assumed between a visual target of interest and the pupil dilation. Therefore, it is interesting to inspect and observe how the pupil dilation of a user is fluctuating, when she/he looks at an item of interest. Given the fact that such behaviour typically takes place during a fixation, we should observe the values of pupil dilation, when a fixation is identified.

Inspired by the studies that usually compare the average pupil dilation that occurs in a specific area of interest with the average pupil dilation to gain insight regarding the cognitive behaviour of the users (Hess and Polt 1960), we propose to compare the average pupil dilation during a video search session with the pupil dilation reported when viewing a specific shot. Considering again the gaze movements of the users employed in the eye-tracking experiment described in section 0, we provide some insights the pupil dilation behaviour during information retrieval. In an indicative example (Figure 5.2) we observe how the pupil dilation fluctuates, when viewing a relevant shot. At the same time a fixation of around 400ms is identified. In this case we notice that the average



pupil diameter reported during the fixation has been increased by an 18% in comparison to the average pupil diameter of the user during the whole search session. Therefore, we propose to take into account of the pupil dilation of a user during a fixation and enhance the fixation-based features with pupil dilation information.

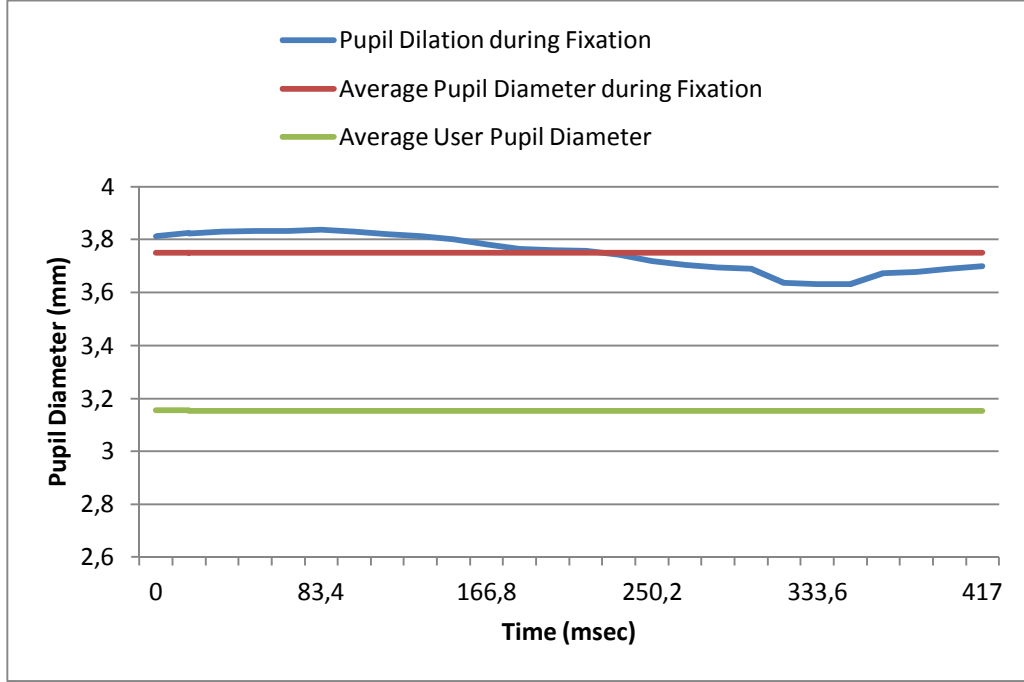


Figure 5.2. Pupil dilation of a user during a fixation

#### 5.2.4. Feature extraction

In this section we propose a feature vector that describes each video shot with respect to the relevance to a specific user query based on eye movement information. The fixation features are based on (Klami, et al. 2008) and (Zhang, et al. 2010), from which we adopt the fixation total duration, the number of fixations and the average duration time and we enhance them by considering relative fixation features (i.e. with respect to the search session duration). On the other hand the pupil dilation features are inspired by (Privitera, et al. 2008), in which we consider pupil dilation information in terms of normalised diameter and speed during the “critical time” that is in our case the fixation time window. In the sequel, we will present in more detail the feature used in this approach.

In order to formally declare the eye movement-based features, we introduce some basic definitions. First, we define as search session  $S_{j,k}$  the time period, during which, user  $j$  is searching for a specific topic  $k$ . We assume that each search session  $S_{j,k}$  lasts  $t_{S_{j,k}}$  time. We declare as  $F_{\alpha,S_{j,k}}$  the total number of fixations and  $T_{\alpha,S_{j,k}}$  the total fixation duration time that were reported for a shot  $\alpha$  during a search session  $S_{j,k}$ . During each fixation time window  $T_{\alpha,S_{j,k}}$  a fluctuation of the pupil data diameter takes place, which is represented by a series of pupil diameter values, sampled with a specific frequency. For each fixation we extract a normalised average diameter value (i.e. the average diameter value reported for this fixation divided by the overall average value of the same user). Using the pupil diameters reported by long search sessions for each user we extract an average pupil diameter value for each eye of each user. We declare as  $D_{R,j}$  and  $D_{L,j}$  the average pupil diameter values for the right and the left eye for user  $j$ . Then for each fixation  $x$  we calculate the average pupil diameter  $D_{x,R,j}$  and normalise against the  $D_{R,j}$ . In parallel we calculate the speed (i.e. rate of change in time) of the pupil dilation. Then the average speed is calculated for each fixation. In case more than one users are considered the aforementioned values are aggregated.

Assuming that we want to describe a shot  $\alpha$  with information retrieved during a set of sessions  $Y = \{S_{j,k}\}$ , where  $j, k \in \mathbb{N}, 0 < j \leq L, 0 < k \leq K$ , where  $L$  is the number of different topics and  $K$  the number of users involved in these sessions. Since we consider that the gaze input could be a result either from one user or aggregated information by many users, the proposed features need to be normalised against the number of users  $K$  (i.e. the features are divided with the number of users) and the number of topics  $L$ . The features and the corresponding mathematical formulas are described in Table 5.1. Hence, the final feature vector for shot  $\alpha$  would be:  $f_\alpha = [F_\alpha, T_\alpha, A_\alpha, V_\alpha, M_\alpha, D_{R,\alpha}, D_{L,\alpha}, S_{R,\alpha}, S_{L,\alpha}]$ .

### 5.2.5. Classification using support vector machines

In order to perform classification, we apply the Support Vector Machines (SVM) algorithm (Boser, et al. 1992) since SVMs have been applied successfully on

several relevant classification problems (e.g. (Jiang, et al. 2010), (Ballan, et al. 2010), (Yildizer, et al. 2012)).

**Table 5.1. Eye movement-based features**

#	Feature description	Mathematical Formula
1	Total number of Fixations for shot $a$	$F_a = \frac{\sum_{S_{j,k} \in Y} F_{a,S_{j,k}}}{L \cdot K}$
2	Total fixation time for shot $a$	$T_a = \frac{\sum_{S_{j,k} \in Y} T_{a,S_{j,k}}}{L \cdot K}$
3	Average fixation time for shot $a$	$A_a = \frac{T_a}{F_a} = \frac{\sum_{S_{j,k} \in Y} T_{a,S_{j,k}}}{\sum_{S_{j,k} \in Y} N_{a,S_{j,k}}}$
4	Average fixations for shot $a$ per search session	$V_a = \frac{F_a}{\sum_{S_{j,k} \in Y} t_{S_{j,k}}} = \frac{\sum_{S_{j,k} \in Y} F_{a,S_{j,k}}}{\sum_{S_{j,k} \in Y} t_{S_{j,k}}}$
5	Average fixation time for shot $a$ per search session	$M_a = \frac{T_a}{\sum_{S_{j,k} \in Y} t_{S_{j,k}}} = \frac{\sum_{S_{j,k} \in Y} T_{a,S_{j,k}}}{\sum_{S_{j,k} \in Y} t_{S_{j,k}}}$
6	Average Normalised Right Pupil diameter	$D_{R,a} = \frac{\sum_{S_{j,k} \in Y} \frac{D_{R,a,S_{j,k}}}{D_{R,j}}}{\sum_{S_{j,k} \in Y} F_{a,S_{j,k}}}$
7	Average Normalised Left Pupil diameter	$D_{L,a} = \frac{\sum_{S_{j,k} \in Y} \frac{D_{L,a,S_{j,k}}}{D_{L,j}}}{\sum_{S_{j,k} \in Y} F_{a,S_{j,k}}}$
8	Average Right pupil dilation speed	$U_{R,a} = \frac{\sum_{S_{j,k} \in Y} U_{R,a,S_{j,k}}}{\sum_{S_{j,k} \in Y} F_{a,S_{j,k}}}$
9	Average Left pupil dilation speed	$U_{L,a} = \frac{\sum_{S_{j,k} \in Y} U_{L,a,S_{j,k}}}{\sum_{S_{j,k} \in Y} F_{a,S_{j,k}}}$

In this work, we propose to employ such a SVM implementation in order to classify the viewed items according to the user interest exploiting the gaze-based

feature vector. More specifically, we make use of the LIBSVM library (Chang and Lin 2001) and we consider a C-Support Vector Classification. Given as training vectors the fixation-based features  $f_i \in R^9$ ,  $i = 1, \dots, l$ , in two classes and a vector  $y \in R^l$  such that  $y_i \in (1, -1)$ , C-SVC (Boser, et al. 1992), (Vapnik, et al. 1995) solves the following primal problem:

$$\min_{w,b,\xi} \frac{1}{2} w^T w - v\rho + \frac{1}{l} \sum_{i=1}^l \xi_i \quad (5.1)$$

subject to:

$$y_i(w^T \phi(f_i) + b) \geq \rho - \xi_i \quad (5.2)$$

where  $\xi_i > 0$ ,  $i = 1, \dots, l$ ,  $\rho \geq 0$ .

The dual is:

$$\min_{\alpha} \frac{1}{2} \alpha^T Q \alpha \quad (5.3)$$

subject to:  $0 \leq \alpha_i \leq \frac{1}{l}$ ,  $i = 1, \dots, l$ ,  $e^T \alpha \geq v$ ,  $y^T \alpha = 0$ , where  $e$  is the vector of all ones,  $C > 0$  is the upper bound,  $Q$  is an  $l$  by  $l$  positive semidefinite matrix,  $Q_{i,j} = y_i y_j$  and  $K(f_i, f_j) = \phi(f_i)^T \phi(f_j)$  is the kernel. In this implementation we consider as kernel the radial basis function (Buhmann, 2003):

$$K(f_i, f_j) = e^{-\gamma |f_i - f_j|^2} \quad (5.4)$$

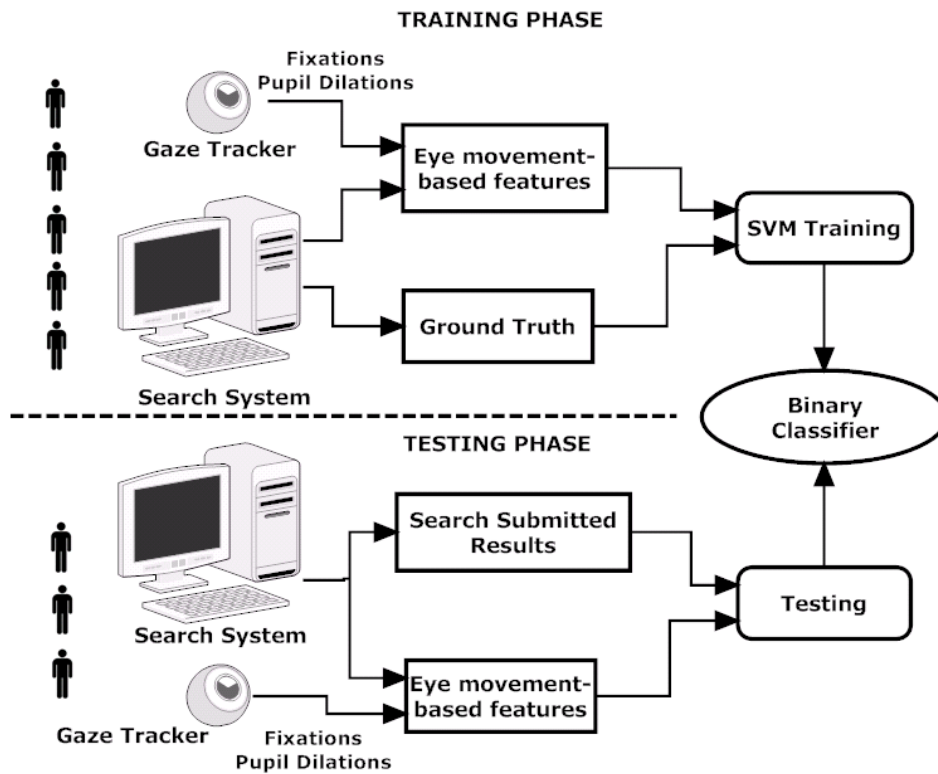
where  $f_i$  are feature vectors of input data  $i$ ,  $f_j$  the support vectors and  $\gamma$  is a constant parameter.

### 5.3. Experiments

#### 5.3.1. Experimental setup

To apply the proposed methodology, we conducted a realistic interactive video retrieval experiment, in which different users searched with the aforementioned video search engine. The experiment took place in the laboratories of Queen

Mary, University of London and 8 subjects (4 male and 4 female) were recruited to participate. The participants were mostly postgraduate students or postgraduate researchers with an average age of 30.5 years old. All of them had a very good knowledge of English and a computer science background. In addition most of them had a good understanding of retrieval tasks and were familiar with search engines.



**Figure 5.3. A schematic view of the experiment**

In this task we made use of the TRECVID 2008 test video set which is described in section 0. The following query topics were used in our experiments:

- A. Find shots of one or more people with one or more horses
- B. Find shots of a map
- C. Find shots of one or more people with one or more books
- D. Find shots of food and/or drinks on a table

The task for each user was to search during a time window of 10 minutes per topic and find as many results that satisfy the given topic. Each user searched for all the topics A-D.

Before the experiment, a tutorial session took place, during which the users were getting familiar with the search engine. In order to imitate a real life video retrieval task in a less controlled environment (compared to (Zhang, et al. 2010)), we instructed the users to search as they normally do, that is without making extra effort to focus their gaze on the shots of interest as they were instructed to do in other similar works (e.g. (Zhang, et al. 2010)). The whole experiment was divided into the training and the testing phase, as it is depicted in the schematic view of Figure 5.3.



**Figure 5.4. FaceLAB eye-tracker**

To record the gaze movement of the users we employed a binocular set of 60Hz cameras with Infra-Red filters (Figure 5.4) and the faceLAB 5.0 software package as the eye-tracking technology<sup>12</sup>. This system requires a user calibration phase, which takes less than one minute. It offers an error of less than 0.5 degrees that suggests approximately less than 5mm diversion from the actual gaze point, when the user is looking at the screen from a distance of 50cm. We used the output of the eye-tracker, in order to gain knowledge regarding the coordinates of each user's gaze for a given time. Then, we processed this information to identify eye fixations and pupil dilations on the video shots. We considered as minimum time of 100ms to define a fixation, during which the gaze was stable.

---

<sup>12</sup> <http://www.seeingmachines.com>

At the same time we also recorded the mouse clicks, the keyboard inputs, the queries and the submissions of the users. In order to provide cross-validated results we consider several different splits of the data with respect to the query topics and the users as it is discussed below.

### 5.3.2. Training phase

As training set, we consider the data retrieved by 5 users, who searched for several combinations of the topics A-D. The results submitted by these users constitute an explicit relevance metric with respect to the query topics for all the viewed items. Considering that very high precisions are reported for interactive systems, given the fact that users select a shot only when it is of relevance to the query topic, the submitted shots comprise a very reliable ground truth set for this task.

**Table 5.2. Training cases**

Training Case/Feature Set	Model No	Features
1	0	1-5
2	1-4	1-5
3	5-8	1-7
4	9-12	1-9

As it is shown in Figure 5.3, we train the SVM models using the feature vectors produced by the fixation and pupil dilation data and the ground truth. In order to evaluate the approach, we provide a variety of training cases, in which different combinations of training features and topics are used. More specifically, the following four training cases, which are shown in Table 5.2 are considered: in the first, we train the classifier (model 0 in Table 5.3) by using the features 1-5 (Table 5.1) and the 4 topics (A-D), in the second case we train recursively 4 different classifiers (models 1-4 in Table 5.4) by selecting each time a different combination of the three topics (i.e. (A, B, C), (A, B, D), etc.) and using as vector the 1-5 fixation-based features, while in the third (models 5-8 in Table 5.5) and forth (models 9-12 in Table 5.6) training cases we repeat the scenario of the second training case, but we make use of the features 1-7 and 1-9 respectively. In

all the aforementioned training cases the gaze data from the same five users are used.

As the ground truth data are not balanced (the positive samples were in average about 10% of the total judged samples) we train the models introducing a corresponding weight  $w_n = 1$  for negative and  $w_p = 10$  for positive classes. More specifically, we set the cost parameter  $C$  to  $w_p \cdot C$  and  $w_n \cdot C$  for positive and negative samples respectively.

### 5.3.3. Testing phase

In the testing phase, the remaining 3 users (different from the ones employed in training phase) are recruited to search for the 4 same topics A-D. In a similar way with the ground truth collection, we capture the video shots that these users identify as relevant to each topic. Then, we utilise this information, in order to test the classifier against the actual selections of the users. Based on the four aforementioned training cases, we test the first classifier (i.e. model 0 of the first training case) by considering all the topics A-D, while we test the other 12 models (i.e. the ones trained with the 3 topics combinations), by using gaze data captured only during the retrieval sessions for the remaining topic (e.g. in the case the training was done with topics A,B,C, we test with topic D).

## 5.4. Results and evaluation

In this section we provide results of the experiment for the proposed method. The evaluation is realised by considering quantitative IR metrics, as well as by presenting a visual view of the shots of interest identified for each topic.

### 5.4.1. Quantitative evaluation

We evaluate the proposed system in two dimensions. First, we investigate the performance of the classifier considering different set of gaze features. Then, we attempt to assess the usefulness of the aggregation by considering data of single users.



In this context we report the classification accuracy, the precision, the recall and the F-Score over the items returned by the system as positive results. During testing the submitted results by the test users form the golden set that is used for the evaluation. Formally, assuming that the classifier returns  $TP$  true positives,  $TN$  true negatives,  $FP$  false positives and  $FN$  false negatives for a topic calculated against the  $V$  positive and the  $N$  negative user selections, the accuracy is computed as  $A = \frac{TP+TN}{V+N}$ , the precision as:  $P = \frac{TP}{TP+FP}$ , and the recall as:  $R = \frac{TP}{V}$ . Then the F-Score is calculated as:  $F - Score = \frac{2PR}{P+R}$ . It is important to consider also IR metrics due to the fact that in several cases the data set can be very imbalanced (e.g. contain many negatives). For instance in the case that 90% of the data are negative samples, by marking all the samples as negatives we will achieve an accuracy of 90%, which however is not satisfactory since no shots of interest would have been identified.

#### 5.4.1.1. Feature performance

With a view to evaluating performance of the gaze-based features we consider the four different training-testing cases described in Table 5.2. For cross-validation purposes we provide in this section the average results after calculating the metrics for each user data combination, in which the data of 5 users are used for training and data of the 3 remaining for testing. Overall 56 different user combinations are considered.

The results for the first and second aforementioned training/test cases, in which the 5 fixation-based features are employed, are reported in Table 5.3 and Table 5.4.

**Table 5.3. First case (features 1-5)**

Model No	Train Topics	Test Topics	Classifier Performance	Precision	Recall	F-Score
0	A,B, C,D	A, B, C, D	94.17%	49.2%	61.95%	54.84%

Starting by observing the results of the first case, it can be concluded that the results for model 0 are of good quality, given the fact that the accuracy is almost 95%, while the F-Score reaches 55%. However it should be noted that, since the

topics A-D are involved both in the training and testing procedure, the results might be biased and therefore a topic independent evaluation should take place.

**Table 5.4. Second case (features 1-5)**

Model No	Train Topics	Test Topics	Classifier Accuracy	Precision	Recall	F-Score
1	B,C,D	A	85.70%	15.04%	88.1%	25.69%
2	A,C,D	B	93.5%	37.16%	61.11%	46.22%
3	A,B,D	C	70.43%	19.52%	89.13%	32.03%
4	A,B,C	D	72.17%	14.32%	73.41%	23.97%
Average			80.45%	21.51%	77.94%	33.72%

**Table 5.5. Third case (features 1-7)**

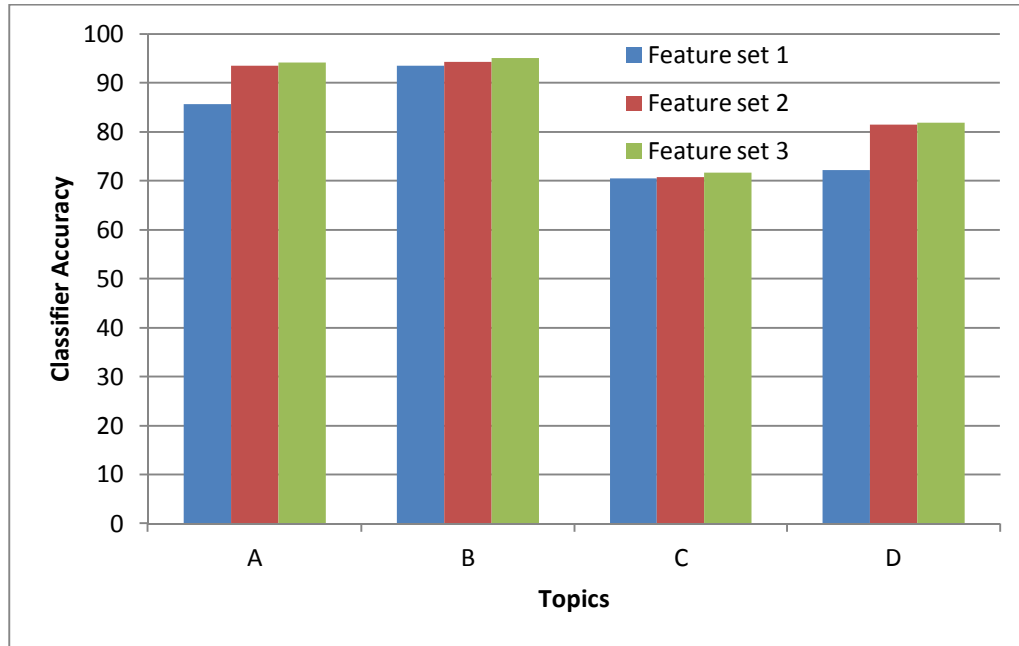
Model No	Train Topics	Test Topics	Classifier Accuracy	Precision	Recall	F-Score
5	B,C,D	A	93.5%	24.6%	73.8%	36.9%
6	A,C,D	B	94.34%	33.6%	45.56%	38.68%
7	A,B,D	C	70.69%	19.86%	90.17%	32.55%
8	A,B,C	D	81.48%	19.51%	70.89%	30.6%
Average			85%	24.39%	70.1%	36.18%

Such an evaluation is realised in the more realistic second case (Table 5.4, models 1-4). The results are still satisfactory, since the average accuracy surpasses 80%. This shows that the proposed method can provide quality results without depending on the topic.

Then, the results for the third and forth training cases are presented in Tables 5.5 and 5.6 respectively.

**Table 5.6. Forth case (features 1-9)**

Model No	Train Topics	Test Topics	Classifier Accuracy	Precision	Recall	F-Score
9	B,C,D	A	94.12%	25.21%	69.05%	36.94%
10	A,C,D	B	95.11%	35.19%	42.22%	38.39%
11	A,B,D	C	71.65%	20.39%	90.22%	33.26%
12	A,B,C	D	81.88%	20.14%	72.15%	31.49%
Average			85.7%	25.23%	68.71%	36.9%

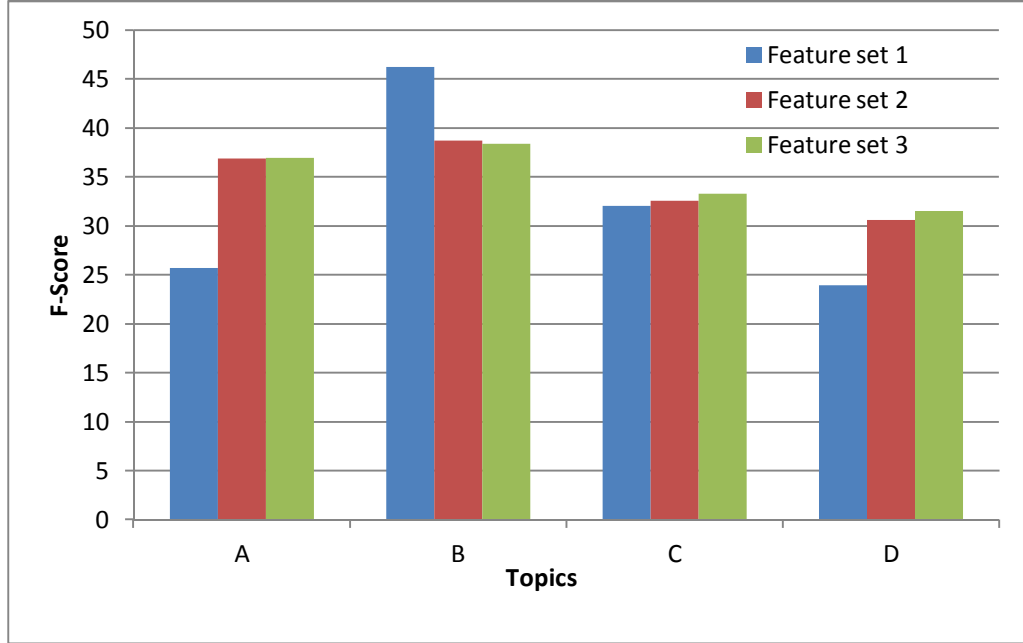


**Figure 5.5. Accuracy of the classifier for topics A-D for the 3 feature sets**

With a view to evaluating the different set of features, we observe that when pupil dilation information is involved, the accuracy of the classifier is slightly improved for all cases. The comparison of the accuracy of the classifiers for the different feature sets is illustrated in Figure 5.5. The major improvement is reported in the testing of topic D (i.e. models 4, 8, 12). In this case, the accuracy of the classifier is boosted from 72.17% to 81.48%, reporting an improvement of 12.9%, when the second feature set (i.e. features 1-7) is involved and a total increase by 13.45%, when we employ the third feature set (i.e. features 1-9). Furthermore, as it is shown in Tables 5.4, 5.5 and 5.6, the involvement of pupil features improves the precision of the system by an average of 13.4%, when we employ features 1-7 and by a further 3.45%, when the third feature set (i.e. features 1-9) is involved. On the other hand, the recall seems to drop by 10% and an additional 1.98% for the two aforementioned cases. The average F-score is calculated as 33.7%, 36.18% and 36.9% for feature sets 1, 2 and 3 respectively, showing that the overall performance of the system slightly improves with the employment of pupil dilation information. Similar, the F-Score is improved from the initial 24% to a final 31.5% when the pupil dilation features are considered.

In Figure 5.6 we present the F-Score for each topic. In the three topics (A, C, D) it is clear that the pupil dilation employment improves the performance. However

the results of topic B show that fixation features perform better, since the F-score drops when the pupil dilation is involved.



**Figure 5.6. Accuracy of the classifier for topics A-D for the 3 feature sets**

After observing precision and recall metrics, it is clear that although the recall values are satisfying, the precision remains rather low (especially in models 1 and 4). In the case that we want to increase the precision at the cost of reducing the recall, we adjust the weighting  $w_n$  and  $w_p$  parameters during the training accordingly. In Figure 5.7, the Precision-Recall curve for model 1 is illustrated, in the case that the ratio  $\frac{w_p}{w_n}$  (points on the Precision-Recall curve) takes values from 0.2 to 10.

#### 5.4.1.2. Evaluating gaze data aggregation

The proposed method is based on feature aggregation by several users. However it is interesting to investigate whether such an aggregation improves the results when compared to the gaze movements of a single user. In order to show how the results are affected, when considering aggregated user gaze data, we investigate the performance of the system in the case that one, two and three users are employed. Further extending the training case 1, we consider as training data the aggregated input by users 1-5 (i.e. the first 5 users), and we attempt to extract

shots of interest for a new user (i.e. the user 6, 7 and 8), who searches for different topics. These results are shown in Table 5.7.

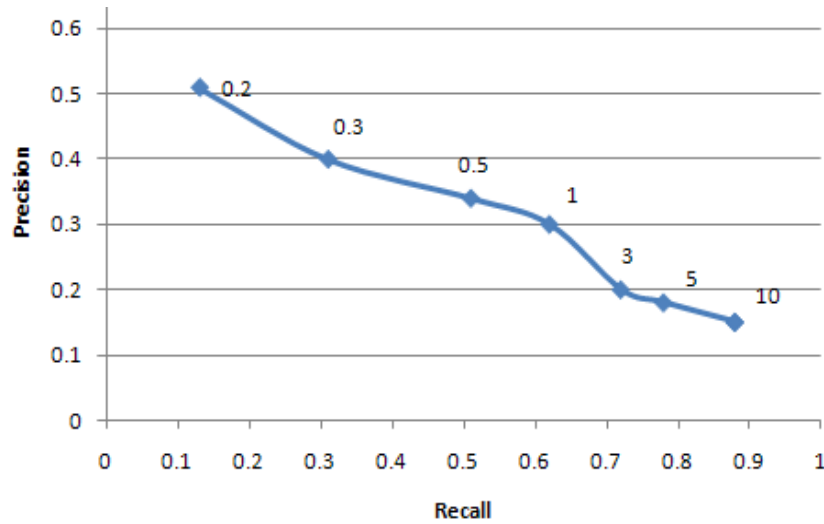
**Table 5.7. Performance of the classifier for each user**

Model	Train Topics	Test Topic	Train Users	Test User	Classifier Accuracy	Precision	Recall	F-Score
1	B,C,D	A	Aggreg. 1,2,3,4, 5	6	83.84% (498/594)	9.2%	60%	15.8%
2	A,C,D	B	Aggreg. 1,2,3,4, 5	6	85.60% (333/389)	59.7%	54.4%	56.9%
3	A,B,D	C	Aggreg. 1,2,3,4, 5	6	69.93% (100/143)	32.8%	82.6%	46.9%
4	A,B,C	D	Aggreg. 1,2,3,4, 5	6	44.44% (24/54)	20.6%	70%	31.8%
1	B,C,D	A	Aggreg. 1,2,3,4, 5	7	39.97% (269/673)	5.4%	92%	10.2%
2	A,C,D	B	Aggreg. 1,2,3,4, 5	7	70.67% (530/750)	7.6%	90%	14.1%
3	A,B,D	C	Aggreg. 1,2,3,4, 5	7	58.16% (410/705)	12.9%	89.6%	22.6%
4	A,B,C	D	Aggreg. 1,2,3,4, 5	7	50.66% (384/758)	7.0%	93.3%	13.1%
1	B,C,D	A	Aggreg. 1,2,3,4, 5	8	74.24% (343/462)	17.4%	100%	29.6%
2	A,C,D	B	Aggreg. 1,2,3,4, 5	8	82.86% (382/461)	23.4%	75.9%	35.8%
3	A,B,D	C	Aggreg. 1,2,3,4, 5	8	67.32% (344/511)	21.2%	86%	34.0%
4	A,B,C	D	Aggreg. 1,2,3,4, 5	8	67.82% (373/550)	18.6%	65.5%	29.0%

Then, in Table 5.8 we present the average results for each of the different training topics by averaging the performance for the 3 users.

**Table 5.8. Average results per topic for training case 1**

Model	Train Topics	Test Topic	Train Users	Test User	Cl. Accur.	Precision	Recall	F-Score
1	B,C,D	A	Aggreg. 1,2,3,4, 5	Aver. 6,7,8	66.01%	10.6%	84.0%	18.9%
2	A,C,D	B	Aggreg. 1,2,3,4, 5	Aver. 6,7,8	79.71%	30.3%	73.4%	42.8%
3	A,B,D	C	Aggreg. 1,2,3,4, 5	Aver. 6,7,8	65.14%	22.3%	86.1%	35.4%
4	A,B,C	D	Aggreg. 1,2,3,4, 5	Aver. 6,7,8	54.31%	15.4%	76.3%	25.6%
Total Average Values					66.28%	19.6%	79.9%	31.5%



**Figure 5.7.** Precision-Recall curve for model 1 when the ratio  $w_p/w_n$  (points on the P-R curve) takes values from 0.2 to 10.

We also present the average results for each user, when averaging the testing results for each topic (Table 5.9). Finally we assess how the performance is improved, when more users are considered. Specifically we report the performance in the case of one, two and three test users searching for topic B (i.e. model 2). These results are illustrated in Table 5.10.

**Table 5.9.** Average results per user for training case 1

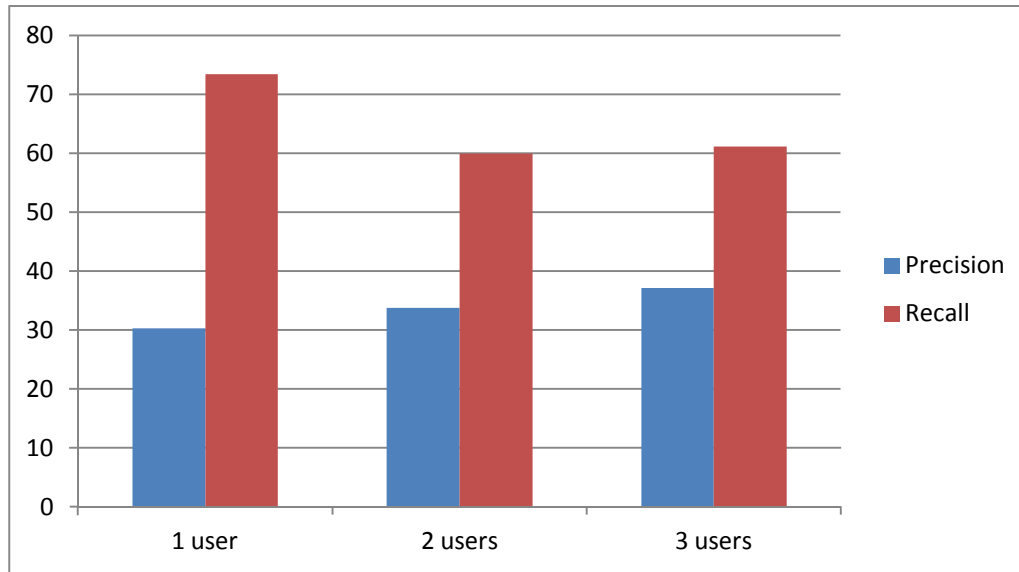
Train Topics	Test Topic	Train Users	Test User	Cl. Accur.	Pre-cision	Recall	F-Score
3 of (A,B,C,D)	1 of (A,B,C,D)	Aggreg. 1,2,3,4, 5	6	70.95%	30.5%	66.8%	41.9%
3 of (A,B,C,D)	1 of (A,B,C,D)	Aggreg. 1,2,3,4, 5	7	54.86%	8.2%	91.2%	15.1%
3 of (A,B,C,D)	1 of (A,B,C,D)	Aggreg. 1,2,3,4, 5	8	73.06%	20.1%	81.8%	32.3%
Total Average Values				66.28%	19.6%	79.9%	31.5%

As far as the precision is concerned, the initial precision reported for the one user is increased by a 3.4% with the involvement of a second user, followed by a further increase of 3.5% in the case that three users are considered. On the other hand, we report a drop of 13.5% in the initial recall (i.e. in the case of the one

user) when aggregated results of two users are considered and slightly increased by a 1.2% when the third user is involved. Finally the F-Score increases from an initial 42.8% to a 43.1%, when two users are involved and reaches the 46.2% for three users. The precision and recall for these cases are illustrated in Figure 5.8, while the F-score is presented in Figure 5.9.

**Table 5.10. Model 2 when data of one, two and three users are aggregated**

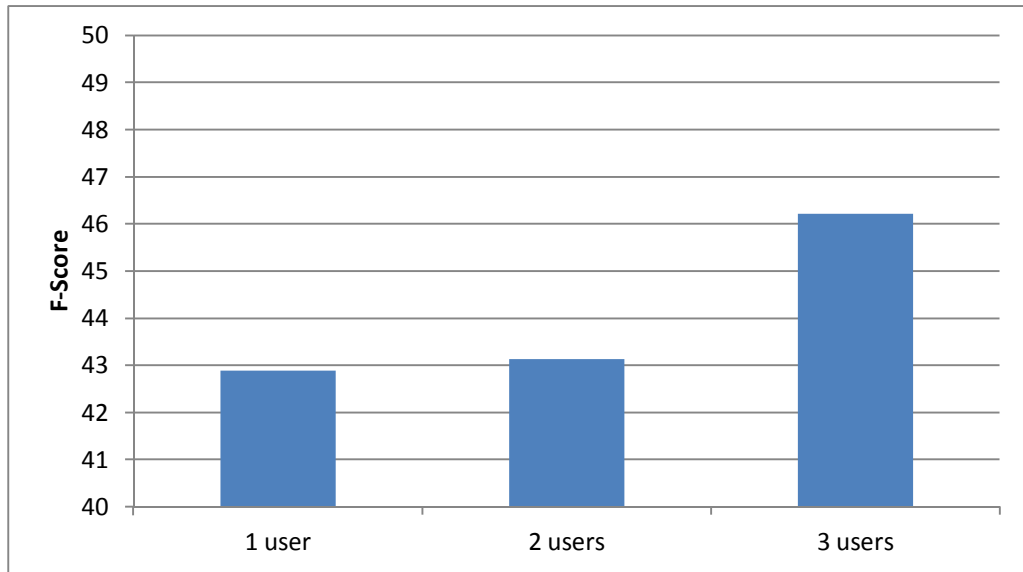
Train Topics	Test Topic	Train Users	Test User	Cl. Accur.	Precision	Recall	F-Score
A,C,D	B	Aggreg. 1,2,3,4, 5	Aver. 6,7,8	79.7%	30.3%	73.4%	42.8%
A,C,D	B	Aggreg. 1,2,3,4, 5	Aggr. Aver. 2 users	86.3%	33.7%	59.9%	43.1%
A,C,D	B	Aggreg. 1,2,3,4, 5	Aggr. 6,7,8	93.5%	37.2%	61.1%	46.2%



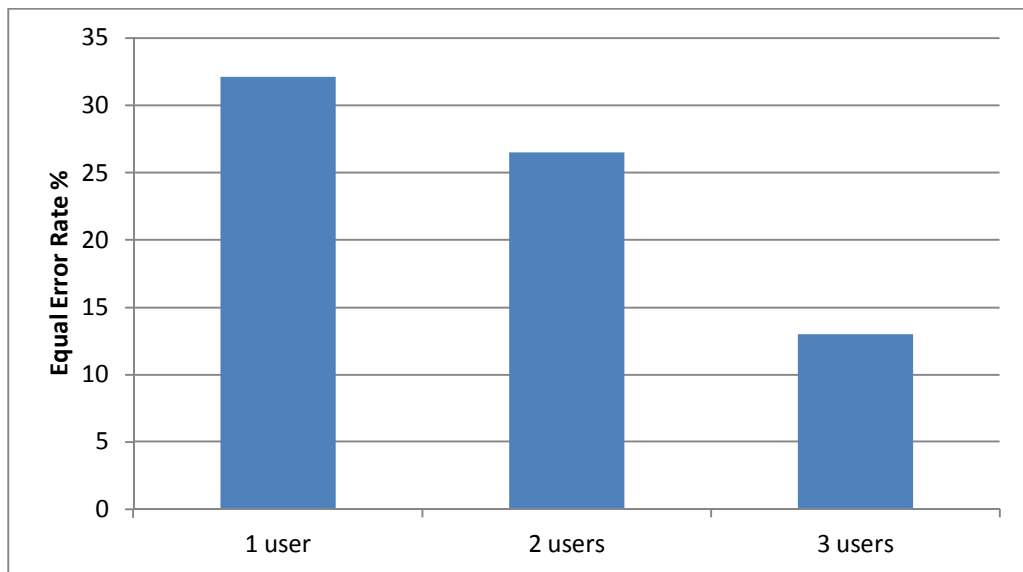
**Figure 5.8. The precision and recall for model 2 are presented, when 1, 2 and 3 users are considered**

We also report the fluctuation of the *EER* in the case of one, two and three test users searching for topic B (i.e. model 2). As it is observed in Figure 5.10, it is clear that the performance of the classifier is improved, as lower values of *EER* are reported for the 3 users. More specifically, the *EER* calculated in the case of one user (i.e.  $EER = 32.1\%$ ), is decreased by 19% when data from a second user are considered, and it is reduced by a further 50% to get a final value of  $EER =$

13.2%, in the case of three users. In a similar way, we present in Figure 5.8 how precision and recall are changing in the evaluation of model 2, when one, two and three test users are involved.



**Figure 5.9. The F-Score for model 2 are presented**

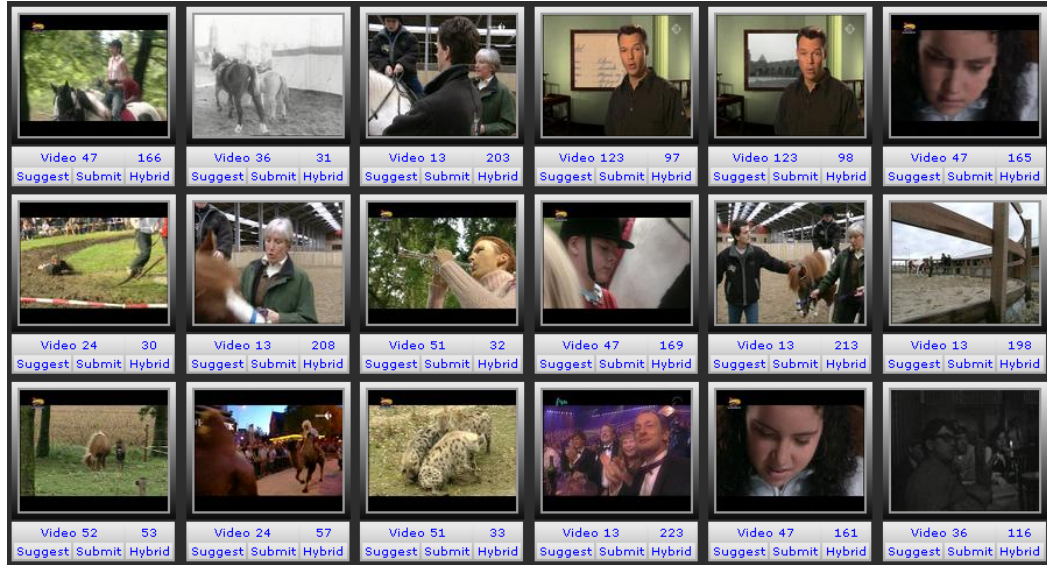


**Figure 5.10. The EER for model 2 is reported, when we use aggregated data from 1,2 and 3 test users respectively.**

This analysis shows that when aggregated gaze information is taken into account the unique gaze behaviours of each user seem to be smoothed to an average gaze behaviour, for which the classifier yields better results. The fact that the classifier's performance improves, when the number of the users involved is



increased, is an indicator that such an approach could be applied for generating recommendations based on past user aggregated gaze data.



**Figure 5.11. Shots of interest for topic A (Find shots of one or more people with one or more horses)**

#### 5.4.2. Visual assessment of results

In order to output shots of interest for a specific query (i.e. topic), the system utilises the classifier output, which is expressed as a distance from the hyperplane that discriminates the two different classes and ranks the shots accordingly. A visual illustration of shots of interest for topics A, B, C and D is provided in Figures 5.11, 5.12, 5.13 and 5.14 respectively. In Table 5.11 we present the precision at the first 18 results  $P@18$ , which are illustrated in the aforementioned figures.

**Table 5.11.  $P@18$  for the topics A-D**

Topic	<i>Precision@18</i>	
Topic A	9/18	0.5
Topic B	14/18	0.778
Topic C	11/18	0.61
Topic D	15/18	0.83
Average	0.68	

In average the precision at the first 18 results reaches 68%. It is interesting to notice that the worst results (50%) are reported for topic A, which is probably

due to the fact that this topic is considered rather difficult, since the users were not able to find many results and therefore probably more fixations are identified to irrelevant shots. On the other hand the topic D achieves a very satisfactory precision of 83%.



Figure 5.12. Shots of interest for topic B (Find shots of a map)

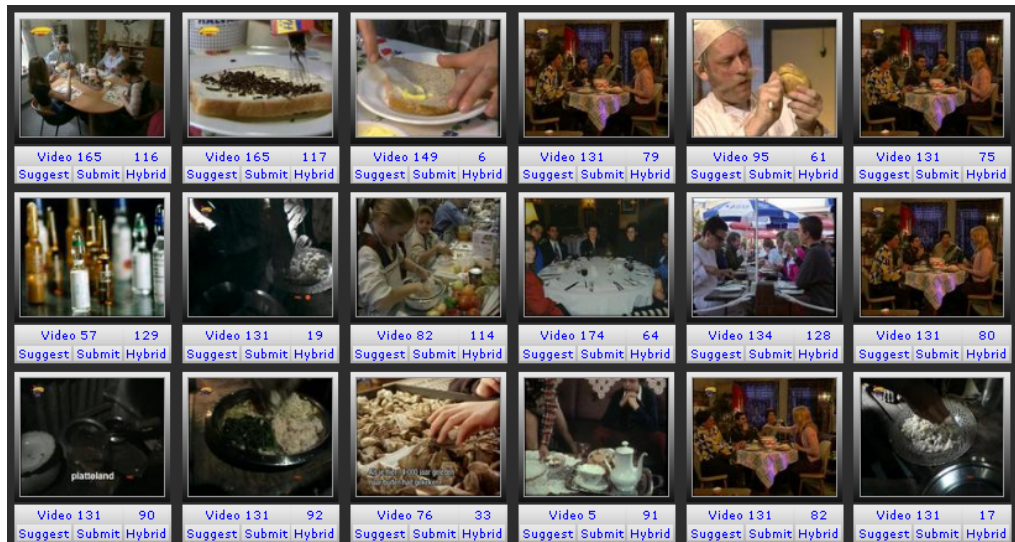
## 5.5. Conclusions

In this chapter we have investigated the role of gaze movements during interactive video retrieval tasks in terms of assisting in the discrimination between relevant and non relevant shots to a given query topic with the aid of SVM classifiers. Our results show that exploiting gaze-based implicit feedback could be of added value in interactive video retrieval tasks as important information regarding the relevance of a video shot to a query topic can be generated even in not controlled environments. After having experimented with a different number of eye movement-based features it seems that pupil dilation information can complement the fixation data in the task of identifying items of interest in the context of a submitted query. In addition, it seems that the usage of aggregated gaze data improves the performance of the system as the precision and recall are improved when the number of the users considered is increased.



**Figure 5.13. Shots of interest for topic C (Find shots of one or more people with one or more books)**

The capability of detecting the user interest in the context of a specific query shows that such an approach could efficiently support an automatic video annotation and tagging system, which could associate search topics with shots of interest. This conclusion comprises the fundamental idea for the research conducted in Chapter 6, in which an automatic annotation framework based on gaze movements and query clustering is proposed.



**Figure 5.14. Shots of interest for topic D (Find shots of food and/or drinks on a table)**

## **Chapter 6**

### **AUTOMATIC VIDEO ANNOTATION BASED ON QUERY CLUSTERING AND EYE MOVEMENTS**

*This chapter proposes a framework for automatic video annotation by performing query clustering and exploiting gaze movements during interactive video retrieval tasks. In this context, we use a content-based video search engine to perform interactive video retrieval, during which, we capture the user eye movements with the aid of an eye-tracking device and record user actions, such as the mouse clicking and queries submission. We use this information to generate feature vectors, which are used to train a classifier that could identify shots that are relevant to new search topics. The queries submitted by new users are clustered in search topics and the viewed shots are annotated as relevant or non-relevant to the topics by the classifier. Query clustering is performed with two different approaches. First we exploit the temporal information and we apply dominant set clustering based on WordNet similarity of textual queries enhanced by the temporal dimension. The second algorithm follows a more sophisticated approach, in which unsupervised random forests utilise gaze movement information, as well as textual and visual query similarity. The experimental results show that gaze movement data can be utilised effectively for automatic video annotation purposes.*

#### **6.1. Introduction**

In the previous decades, manual annotation of multimedia was one of the usual practices to support video and image retrieval. However, in the recent years, the rapid increase of the amount of content has made such practices very costly in terms of human effort. To this end, several automatic annotation approaches have

been devised. Most of them are based on the extraction of low-level features and the generation of high-level concepts (e.g. (Zhang, et al. 2008), (Mezaris, et al. 2010)), facing however the well known problem of semantic gap. More recently, the high availability of sensors, which allowed for the generation of a plethora of behavioural and user interaction data, directed the research trends (e.g. (Vrochidis, et al. 2011), (Tsikrika et al. 2009)) to move towards exploiting the implicit user feedback for image and video tagging and annotation purposes.

In parallel, the plethora of the user interaction data and search logs have motivated several works to focus on query clustering in order to support automatic annotation systems and provide query expansion and recommendation options. Grouping together queries with strong semantic relations is a task that is intrinsically harder than classic topic extraction or document clustering (Zeng, et al. 2004), because of the limited textual information contained into queries. Most of the approaches dealing with query clustering rely on the computation of similarity metrics between query pairs. When dealing with query classification, where the semantic categories are predefined, it may be sufficient to compute the similarity based only on textual features to obtain good classification results (Beitzel, et al. 2007). However, if predefined categories are not available, lexical and content-based information taken separately are not sufficient to obtain good clusters. In this context, an attempt to cluster queries from the Encarta user logs (Wen, et al. 2002) showed that query-to-query similarity metrics that linearly combine textual features with click-through data can be used much more profitably in query clustering than single-attribute similarities. Inspired by such approaches, this work proposes to enhance textual and content-based query clustering by considering implicit user feedback expressed as click-throughs and gaze movements.

In this context, we propose an automatic annotation system, which considers a more realistic retrieval scenario (compared to the one discussed in Chapter 5) by assuming that the search topics during testing may be not only different than the ones encountered during training, but also unknown. The idea is to cluster the submitted queries into groups and then aggregate the user implicit feedback expressed by query submissions and gaze movements across these clusters and

subsequently identify relevant shots for these clusters by exploiting the methodology of Chapter 5 (Vrochidis, et al. 2011). Specifically, we attempt to identify and label unknown search topics by employing a variety of clustering techniques instead of considering known topics, as it was assumed in the previous chapter. During training, the aggregated gaze movements of past users are processed, in order to extract fixations (i.e. spatial stable gaze). Then, we extract a set of features that describes each video shot based on fixation characteristics. Subsequently, we employ a Support Vector Machine (SVM) approach to train a binary classifier that could predict, which of the items viewed by a new user could be classified as interesting for her/him and apparently matches the topic she/he searches for. During testing we assume that we have no knowledge of the topics the user searches for. To identify the unknown topics, we consider two different approaches. The first approach performs dominant set clustering based on WordNet distance of the submitted textual queries and temporal information. The second approach proposes a more sophisticated methodology, in which the clustering algorithm is driven by the performance of the classifier, which depends on the gaze movements. In this case, unsupervised random forests are employed. In contrast to the first approach, in which the SVM classifier that predicts the relevance of the shots with respect to the query clusters, is employed only when the clusters are finalised, in the second approach, the quality of the separation achieved by the classifier is taken into account during the clustering process in order to optimise it. After the clusters have been defined, the positive results of the SVM are associated with the cluster labels, which derive from the queries, annotating in that way the content.

To evaluate the proposed approach, we exploit the data (i.e. user interaction, implicit feedback) gathered during the video retrieval experiment described in section 5.3.1. Then, we present results for each clustering technique and the accomplished annotations. Finally we provide comparisons between the results with the different clustering techniques.

The novel research contributions of this chapter are summarised in the proposed methodology and framework of automatic annotation using gaze features and query clustering. An important contribution is the approach of gaze driven query

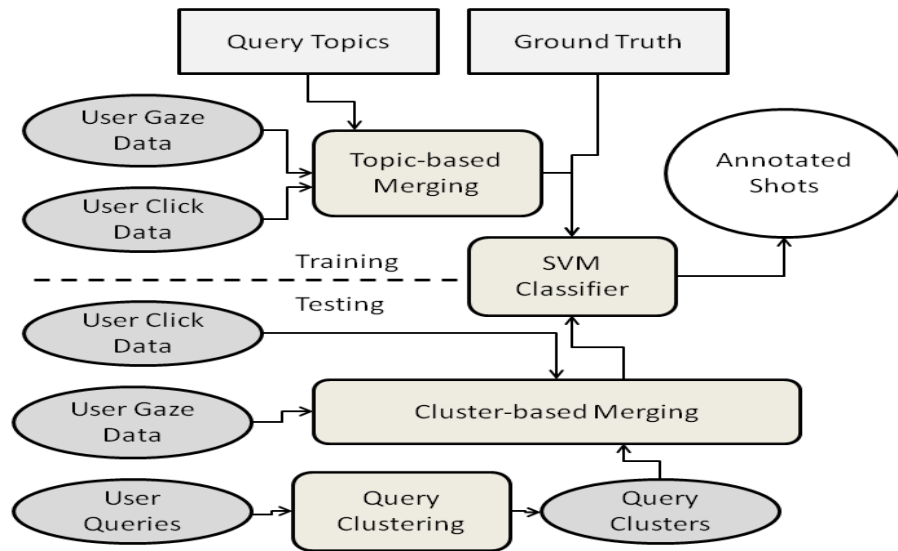


clustering using unsupervised random forests. Parts of this chapter have been published in (Vrochidis, et al. 2012).

This chapter is structured as follows: section 6.2 presents the video annotation framework, while the first clustering approach based on dominant sets is described in section 6.3. In section 6.4, we analyse the query clustering approach based on gaze-driven random forests and the results for both techniques are presented in section 6.5. Finally, section 6.6 concludes this chapter.

## 6.2. Video annotation framework

We consider a supervised machine learning framework (based on classification), which, during the testing phase, includes also an unsupervised learning phase (clustering). The framework is illustrated in Figure 6.1.



**Figure 6.1. Video annotation framework**

During the training phase, we assume we have explicit knowledge of the query topics the users are searching for, how much time they search for each topic and the queries they submit (e.g. for every topic a user could submit several queries), as well as the results they are interested in. In this phase, the gaze movements of the users searching for the same topic are collected and aggregated. Then, gaze-based features for each video shot are extracted. In the following, we use the gaze features and the results for each topic submitted by the users as explicit

relevance of each shot to a topic, in order to train a Support Vector Machine (SVM) classifier that could classify as relevant or non relevant the items viewed by a new user. Since this classifier is trained only with gaze movements, it can be considered as predictor of user interest for a certain viewed item in the context of a query topic.

In the testing phase, we assume that we have no knowledge regarding the query topics (i.e. topic subject, time boundaries). To identify the unknown topics during testing, we perform clustering following two different approaches: a) dominant set clustering using WordNet similarity between textual queries and temporal information, b) gaze driven unsupervised random forests taking into account WordNet distances between textual queries and visual similarity between clicked keyframes, as well as the performance of the SVM classifier, which is driven by the gaze movements. Then, we employ the aforementioned classifier to predict the relevance of the shots with respect to the query clusters. The positive results of the SVM are associated with the cluster labels, annotating in that way the video shots.

### **6.3. Dominant set query clustering using temporal information**

During an interactive video search session many users could search for several topics from the same computer terminal. In most of the cases the queries submitted by the user can give a good idea of what she/he is searching for. However, the user usually queries the system using specific keywords and not the whole query itself. In addition, the user could change arbitrarily the search topic, which might further complicate the situation.

We consider a search session  $S$ , during which,  $K$  users are searching for  $N$  topics  $\{z_1, z_2, \dots, z_N\}$  and they submit  $M$  queries  $\{Q_1, Q_2, \dots, Q_M\}$ . Since we assume that the users are searching in a sequential way (i.e. the after other), the goal is to find the time boundaries for each topic, as well as to define the topics in terms of textual description. We consider that each topic  $z$  is described by a set of queries  $\{\dots, Q_{k-1}, Q_k, Q_{k+1}, \dots\}$ . The timeline of the search session is illustrated in Figure



6.2. Each query  $Q$  can have as input either text or a shot. We declare as  $v_{i,j}$  the semantic distance between two queries  $Q_i$  and  $Q_j$ . The aim is to arrange the queries in such groups for which the  $v_{i,j}$  between the queries of the same group (topic) will be minimised, while the  $v_{i,j}$  between queries belonging in different groups (topics) will be maximum. This can be considered as a clustering problem, in which we want to organise queries into an unknown number of topics considering pairwise similarity and time dimension.

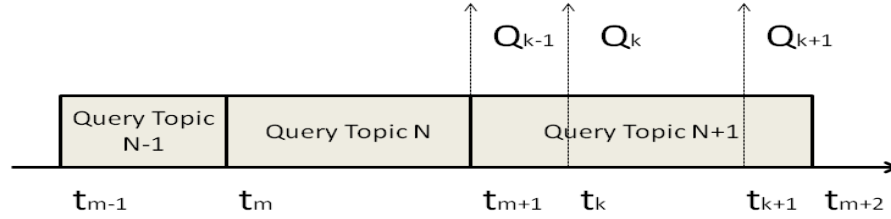


Figure 6.2. Search session and queries

### 6.3.1. Dominant set clustering

Dominant set as defined in (Pavan and Pelillo 2003) is a combinatorial concept in graph theory that generalises the notion of a maximal complete subgraph to edge-weighted graphs. It simultaneously emphasises on internal homogeneity and external inhomogeneity, and thus is considered as a general definition of cluster. The authors in (Pavan and Pelillo 2003) establish an intriguing connection between the dominant set and a quadratic program as follows:

$$\max f(x) = x^T S x, x \in \Delta \quad (6.1)$$

where  $\Delta = \{x \in \mathbb{R}^n : x \geq 0 \text{ and } \sum_{i=1}^n x_i = 1\}$  where  $S$  is the similarity matrix. Specifically, it is proven that if  $S$  is a dominant subset of vertices, then its weighted characteristic vector  $x^S$ , which is the vector of  $\Delta$  defined as:

$$x_i^S = \begin{cases} \frac{w_s(i)}{W(S)}, & \text{if } i \in S \\ 0, & \text{otherwise} \end{cases} \quad (6.2)$$

is a strict local solution of (6.1). Conversely if  $x$  is a strict local solution of the above problem, it is proven by (Pavan and Pelillo 2003) that  $\sigma(x) = \{i | x_i > 0\}$

is equivalent to a dominant set of the graph represented by  $S$ . Then, the replicator equation is used to solve (6. 1):

$$x_i(t + 1) = x_i(t) \frac{(Sx(t))_i}{x(t)^T Sx(t)} \quad (6. 3)$$

**Table 6.1. Dominant set clustering algorithm**

---

Input: the similarity matrix  $S$

1. Initialise  $S^j$ ,  $j = 1$  with  $S$
2. Calculate the local solution of (6. 1) by (6. 3):  $x^j$  and  $f(x^j)$
3. Get the dominant set:  $D^j = \sigma(x^j)$
4. Split out  $S^j$  from  $S^j$  and get a new similarity affinity matrix  $S^{j+1}$
5. If  $S^{j+1}$  is empty, break, else  $S^j = S^{j+1}$  and  $j = j + 1$ , then go to step 2

Output =  $\cup_{l=1}^j \{D^l, x^l, f(x^l)\}$

---

The concept of dominant set provides an effective framework for iterative pairwise clustering, which is required in our problem. Considering a set of samples, an undirected edge-weighted graph is built, in which each vertex represents a sample and two vertices are linked by an edge, the weight of which represents their similarity. To cluster the samples into groups, a dominant set of the weighted graph is iteratively found and removed from the graph until the latter is empty. Table 6.1 outlines the algorithm. The dominant set clustering automatically determines the number of the clusters and has low computational cost.

After we employ the dominant set clustering algorithm and form the clusters, the cluster labels are formed by the most frequent keywords included in the queries that comprise each cluster.

### 6.3.2. Query similarity

As explained in the previous section, in order to identify the topic time boundaries, we propose to compare the queries submitted and identify clusters that correspond to search topics. Based on the analysis we performed in Chapter

4 (Vrochidis, et al. 2011), we can identify autonomous and dependent queries and make the assumption that a topic change takes place only at the autonomous query submission. According to this definition, the autonomous queries do not depend on previous results, while the dependent do. In this clustering approach and in order to simplify the problem, we propose to compute similarities between autonomous queries (i.e. textual queries in our case) and assign cluster labels to them. Given the fact that the autonomous queries contain textual information, we need to model a similarity measure between the queries submitted as keywords. In addition, we need to incorporate the temporal dimension in the similarity metric.

#### **6.3.2.1. WordNet-based similarity**

One of the state of the art techniques for comparing textual information is to use thesaurus such as WordNet<sup>13</sup>. In this work we have applied the WordNet “vector” similarity after experimenting with other WordNet metrics (i.e. lesk and path). Each concept (or word sense) in WordNet is defined by a short gloss. The vector measures use the text of that gloss as a unique representation for the underlying concept. The vector measure creates a co-occurrence matrix from a corpus made up of the WordNet glosses. Each content word used in a WordNet gloss has an associated context vector. Every gloss is represented by a gloss vector that is the average of all the context vectors of the words found in the gloss. Relatedness between concepts is measured by finding the cosine between a pair of gloss vectors (Pedersen, et al. 2004).

An additional problem in our case is the inability of dealing with term disambiguation (since the search topics and the context are considered unknown). To overcome this problem we calculate the maximum similarity between the senses of the two textual queries. Although the lack of this information could lead in many cases to erroneous results, we assume that the temporal information,

---

<sup>13</sup> <http://wordnet.princeton.edu/>

could help in distinguishing irrelevant queries that have been submitted in moments varying in time.

### 6.3.2.2. Temporally enhanced similarity

The aim of query clustering is to temporally segment the search time into sessions, in which a user searches for a specific topic. In this case, not only the query similarity but also the temporal constraint has to be taken into consideration. For this reason, we incorporate the temporal dimension into the computation of the similarity matrix with a Gaussian kernel. Hence, the similarity  $w_{i,j}$  between queries  $i, j$  is computed by:

$$w_{i,j} = v_{i,j} \cdot e^{(-\frac{1}{d}|\frac{t_i - t_j}{\sigma}|^2)} \quad (6.4)$$

where  $v_{i,j}$  is the WordNet similarity between the two queries,  $t_i$  and  $t_j$  are the temporal moments, in which the queries  $i, j$  are respectively submitted,  $\sigma$  and  $d$  are the decay factors, which reflect the decreasing rate of the similarity with the temporal interval increasing and  $w_{i,j}$  correspond to the elements of the final similarity matrix  $S$ .

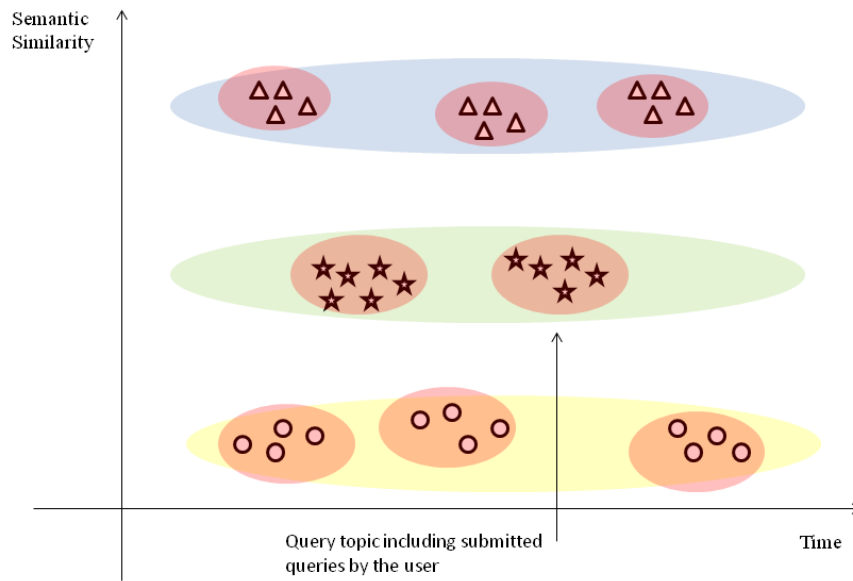
### 6.3.2.3. Smoothing process

We employ the clustering approach to our problem with the assumption that the queries that fall into one cluster constitute a semantic topic. However, there are cases, in which either the user might submit semantically irrelevant queries during a topic search, or the WordNet similarity might not perform very well. Thus, after conducting the standard clustering process, we introduce the following smoothing process:

- a) if the cluster label of a query does not coincide with its two adjacent frames, we assume it was initially misclassified;
- b) small clusters are merged with the adjacent ones. The minimum number of members for defining a small cluster is selected experimentally.

#### 6.4. Query clustering based on gaze-driven random forests

In the previous section, we attempted to perform query clustering taking into account temporal information. Although this information can be very helpful to distinguish temporally neighbouring topics, it could fail in the case that one or more users are searching again for the same topic after a certain time window. In such a case, the previous approach could not be able to group together the queries of these topics and therefore similar topics will be represented by different clusters.



**Figure 6.3. Search sessions by several users in the temporal and semantic similarity dimension. The large clusters indicate semantic topics, while the smaller ones search sessions deal with these topics.**

Therefore, we provide an alternative formulation of the problem in order to overcome this disadvantage. We consider a set of  $N$  search sessions  $S = \{S_1, \dots, S_i, \dots, S_N\}$ . During session  $S_i$  the user  $i$  is searching for a specific topic  $t_i$ . Again, the objective is to aggregate the implicit feedback gathered during by all users, who have searched for topic  $t_i$ . During each session  $S_i$ , the user  $i$  is submitting  $M_i$  queries  $\{Q_1, \dots, Q_{M_i}\}$ . The actual goal in this case is to group the

semantically relevant queries into topics regardless of the time these have been submitted.

Figure 6.3 shows the queries submitted by several users in the context of different topic search topics. The search sessions are organised according to semantic similarity and temporal dimension. As it is illustrated, we assume that there are cases, in which the users are searching again for the same topic. This is shown by the small clusters with similar semantic similarity (i.e. group of queries) that fall into the largest clusters that represent the topics.

In order to perform clustering we need define semantic similarities between the queries. Due to the fact that the temporal information is not taken into account in this case, the problem becomes more challenging in comparison to the one introduced in the previous section, since WordNet similarity would not be adequate to cluster the queries. To this end we consider the following additional information that can be used to form the similarities and drive the clustering process.

Instead of performing a clustering of the autonomous queries, we propose to realise a clustering based on the subsession information and therefore compare both the autonomous and dependent queries. In this context, instead of simply comparing the textual similarities we will consider the images clicked during a subsession, in order to form the dependant queries (e.g. query by visual example). The discussion of the similarity metric is described in detail in section 6.4.3.1.

An additional assumption that is taken into account is that as the cluster converges with the initial topic, the separation of the relevant and irrelevant shots that will be achieved by the interest predictor (i.e. the SVM classifier) should be optimum and therefore the distance of the classified items from the hyperplane would be maximised. In this approach we propose to exploit this assumption to drive the clustering process. We propose to employ unsupervised random forests to perform clustering, since the latter is a convenient and also powerful method that can be applied in this case and incorporate the classification quality of the clusters in the clustering procedure.

### 6.4.1. Random forests

Random Forest is an ensemble classifier that consists of many decision trees and outputs the class that is predicted by the most individual trees. The algorithm for inducing a random forest was proposed by Leo Breiman (L. Breiman 2001). The method combines Breiman's "bagging" idea and the random selection of features, introduced independently in (Ho 1995), (Ho 1998) and in (Amit and Geman 1997), in order to construct a collection of decision trees with controlled variation. The selection of a random subset of features is an example of the random subspace method, which, in Ho's formulation, is a way to implement stochastic discrimination (Kleinberg 1997). Random forests have been used successfully in several classification, regression and clustering problems (e.g. (Shi, et al. 2006), (Bosch, et al. 2007)).

#### 6.4.1.1. Construction of random forests

A random forest may contain hundreds or thousands of trees depending on the application and the dataset. Let us assume that we want to build a forest with  $T$  trees. The following algorithm is used to construct each tree of the forest: we assume that we have  $N$  training examples. Each of them is represented in the multidimensional space by a vector  $v = (x_1, \dots, x_d) \in \mathbb{R}^d$ , where  $d$  is the cardinality of the features employed. Each training example is associated with one of the  $k$  classes  $\{c_1, \dots, c_k\}$ . To proceed with the decision tree construction, we take a *bootstrap sample*, which is used as a training set to grow each tree. This is performed by sampling  $n$  samples with replacement (i.e. by putting back in the collection the selected sample) from all the  $N$  available training cases. The observations that are not in the training set, roughly 1/3 of the original data set, are referred to as out-of-bag (OOB) observations. Assuming that the initial training set is  $S_o$ , we create a training set  $S_o^t \subset S_o$  for each tree  $t$ . In Figure 6.4 we illustrate how the first four trees are grown to construct the random forest.

In order to grow each tree we set a number of  $m < d$  variables, which are considered to randomly select  $m$  of the  $d$  dimensions. In the traditional decision tree construction, all the dimensions of the vector are taken into account and the

best split is decided after comparing the quality of each split using an impurity function. On the contrary, in the case of RF decision tree,  $m$  dimensions are randomly selected and the best split is selected by maximizing an impurity function. Each tree is fully grown and not pruned.

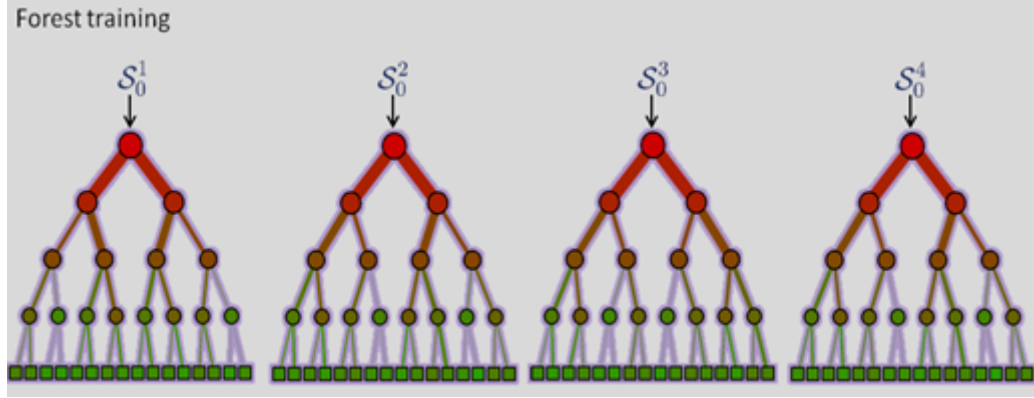


Figure 6.4. Random forest generation

#### 6.4.1.2. Impurity function

The impurity function measures the extent of purity for a region containing data points that possibly belong to different classes. Let us assume that the number of classes is  $K$ . Then, the impurity function is a function of  $p_1, \dots, p_K, \dots, p_K$ , where  $p_i$  is the probability for any data point in the region belonging to class  $i$ . During training, we are not aware of the real probabilities. However, an acceptable compromise is to associate these probabilities with the percentage of points that belongs to each class in the region we are interested in exploiting the labels of the training data set.

Formally, an **impurity function**  $\Phi$  is defined on the set of all  $K$ -tuples of numbers  $(p_1, \dots, p_K)$  satisfying  $p_j \geq 0$ ,  $j = 1, \dots, K$ ,  $\sum_1^K p_j = 1$  and has the following properties:

1.  $\Phi$  achieves maximum only for the uniform distribution of  $p_j$ , which means that all the  $p_j$  are equal.



2.  $\Phi$  achieves minimum only at the points  $(1, 0, 0, \dots, 0)$ ,  $(0, 1, 0, \dots, 0)$ , ...,  $(0, 0, \dots, 0, 1)$ , i.e., when the probability to belong in a certain class is 1 and 0 for all the other classes.
3.  $\Phi$  is a symmetric function of  $p_j$ , i.e., if we permute  $p_j$ ,  $\Phi$  doesn't change.

Given an impurity function  $\Phi$ , we define the impurity measure, denoted as  $i(t)$ , of a node  $t$  as follows:

$$i(t) = \Phi(p(1|t), p(2|t), \dots, p(K|t)) \quad (6.5)$$

where  $p(j|t)$  is the estimated posterior probability of class  $j$  given a point is in node  $t$ . This is called the impurity function (or the impurity measure) for node  $t$ .

Once we have defined  $i(t)$ , we can estimate the goodness of split  $s$  for node  $t$ , denoted by as  $\Delta i(s, t)$ :

$$\Delta i(s, t) = i(t) - w_R i(t_R) - w_L i(t_L) \quad (6.6)$$

$\Delta i(s, t)$  represents the difference between the impurity measure for node  $t$  and the weighted sum of the impurity measures for the right child and the left child nodes. The weights,  $w_R$  and  $w_L$ , are the proportions of the samples in node  $t$  that go to the right node  $t_R$  and the left node  $t_L$  respectively. Two of the most popular impurity functions to base the decision for the best split in each node are the Information Gain and the Gini index (Breiman, et al. 1984).

#### 6.4.1.3. Predicting classes with random forests

After the Random Forest has been constructed, we are able to provide predictions for a test dataset regarding the probability that has each observation belonging to a specific class. Specifically, for each observation, each individual tree votes for one class and the forest predicts the class based on the majority of votes. Formally, assuming that the forest has  $T$  trees and the probability for a new sample  $y$  belonging to class  $c$  as this is predicted by tree  $t$  is  $p_t(c|y)$ , the final probability is:

$$p(c|y) = \frac{1}{T} \sum_1^T p_t(c|y) \quad (6.7)$$

#### 6.4.1.4. Advantages and disadvantages of random forests

Random forests have several advantages. First they are considered as one of the most accurate learning algorithms available, since for many data sets, they produce a highly accurate classifier (Caruana, et al. 2008). Second, they can be easily applied to large databases and in addition can handle thousands of input variables without variable deletion or dimensionality reduction. Another important characteristic of the RF algorithm is that it can compute proximities between pairs of cases that can be used in clustering, locating outliers, or (by scaling) give interesting views of the data. Therefore, the aforementioned advantages can be extended to unlabeled data, leading to unsupervised clustering, data views and outlier detection.

However, it should be noted that there are cases, in which Random forests have been observed to overfit for some datasets with noisy classification/regression tasks.

#### 6.4.2. Unsupervised random forests

Machine learning methods are usually categorised into supervised (annotated examples for training exist) and unsupervised (no labelled data are available) learning methods. Interestingly, many supervised methods can be converted into unsupervised methods using the following idea. An artificial class label is created, which distinguishes the observed data from suitably generated synthetic data. In other words the observed data are labelled with class *A*, while the synthetic with class *B*. The observed data are the original unlabelled data, while the synthetic data are generated from a reference distribution. The supervised learning methods that attempt to distinguish observed from synthetic data by using the aforementioned technique, yield a dissimilarity measure that can be used as input in subsequent unsupervised learning methods. In (Breiman and Cutler 2003) it is proposed to use random forest (RF) predictors to distinguish observed from synthetic data. When such a dissimilarity measure is used as input

in unsupervised learning methods (e.g. clustering) we can generate patterns, which may correspond to clusters in the Euclidean sense of the word. The dissimilarity that results after unsupervised random forest construction has been successfully used in several unsupervised learning tasks involving genomic data. For instance (Breiman and Cutler 2003) applied RF clustering to DNA microarray data, (Allen, et al. 2003) used it to cluster genomic sequence data, while (Shi and Horvath 2006) applied it to tumor marker data. In the following we will discuss in more detail the RF dissimilarity.

#### 6.4.2.1. Random forest dissimilarity

A RF predictor is an ensemble of individual classification tree predictors (Breiman 2001). First we will briefly review how to use random forests to arrive at a dissimilarity measure for labelled data. Since an individual tree is unpruned, the terminal nodes will contain only a small number of observations. The training data are run down each tree. In case two observations  $i$  and  $j$  end up to the same terminal node, the similarity between  $i$  and  $j$  is increased by one. After the forest is finalised, the similarities are normalised and divided by the number of trees. Note that in this way the similarity between an observation and itself becomes one. The similarities between objects form a symmetric matrix, which is positive definite, and each entry lies in the unit interval  $[0\ 1]$ . The RF dissimilarity is mathematically defined as:

$$DS_{ij} = \sqrt{1 - SM_{ij}} \quad (6.8)$$

where  $SM_{ij}$  stands for the similarity between  $i$  and  $j$ .

After having defined the dissimilarity for labelled data, we will review how RF are used to arrive at a dissimilarity measure for unlabelled data (Breiman and Cutler 2003). The idea is to use the similarity matrix constructed from a RF predictor that distinguishes observed from synthetic data. The observed data are the original, unlabelled data, while the synthetic data are drawn from a reference distribution as described in section 6.4.2. By restricting the resulting labelled similarity measure to the observed data, we define a similarity measure between

unlabelled observations. Of course it should be noted that the similarity measure strongly depends on the method for synthetic observations construction.

### 6.4.3. Gaze-driven random forests

In Chapter 5 we have investigated the role of gaze movements and we have created classifiers that can predict interesting shots in a context of a specific topic, when considering aggregated gaze-based features. During query clustering we attempt to cluster the queries in such a way, so that they form the initial query topics. The shots that are viewed in each cluster will be separated by the SVM classifier based on the aggregated features.

We make the assumption that the more the cluster converges to a topic, the best separation is achieved by the SVMs. In the case that the cluster is not well formed, the aggregation of features for shots belonging to different topics will take place and therefore the SVM results will be of low quality. Given the fact that we cannot consider ground truth at this stage, the only indication is the quality of separation that can be revealed by the distances of the classified vectors from the hyperplane. Then, we need to incorporate this criterion in the process of generating the clusters in order to optimise their creation. As we will describe in section 6.4.3.2 this information will be considered during the tree construction procedure and specifically in the definition of the best split.

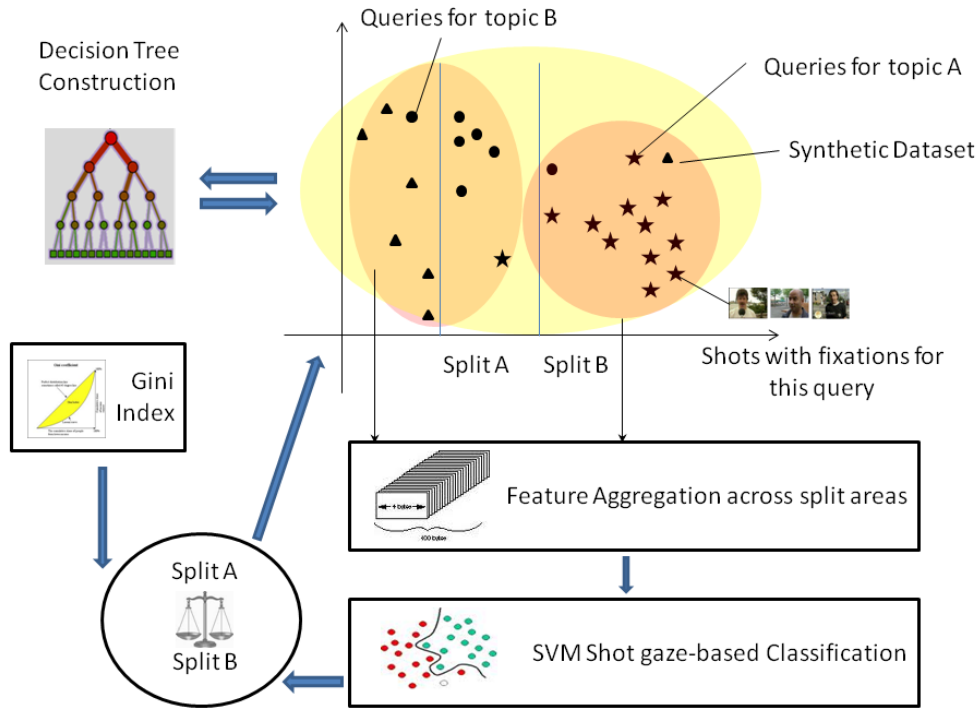
In order to perform random forest clustering we first create an affinity matrix between the submitted queries. Then, we generate the synthetic data and proceed with the creation of random forests. However, the most important part and the novelty of the proposed algorithm is the way we exploit the gaze information.

First we create the synthetic data by randomly sampling from the hyper-rectangle that contains the observed data, i.e. the variables of synthetic observations have a uniform distribution with range determined by the minimum and maximum of the corresponding observed variable.

Then we generate each tree of the forest by considering a different partition of the data ( $\sim 2/3$  of them). The sampling is done in a random way with replacement. Then, we consider  $m$  variables and for each of them we randomly select the

dimension based on which we will perform the splitting. After experimental tuning the  $m$  is selected to be close to  $\sqrt{M}$  where  $M$  is the cardinality of the features for each vector.

We propose the framework in Figure 6.5 for decision tree construction, in which we illustrate the algorithm with an example. To construct a decision tree of the random forests, several splits are constructed and compared. Let's assume that queries for topics A and B are to be clustered. In this example we show the separation in feature  $k$  and two different splits (split A and B) have to be compared. As we have discussed in Chapter 5, in the context of each query several fixations are identified on the resulted shots. In this case, we consider two different aggregations, one by each split. Then, the quality of the classifier separation is incorporated into the splitting criterion to complement the Gini index. The splitting criterion will be discussed in detail in section 6.4.3.2.



**Figure 6.5. Decision tree construction in gaze-driven random forests**

It should be noted that by using the Gini index, the algorithm attempts to separate the synthetic class from the observed values. By incorporating the quality of classification, we also attempt to separate the queries of different topics. For

instance in the demonstrated example, both splits seem to separate synthetic and observed values. However, it seems that split B also manages to separate the queries of different topics and therefore it should have been preferable.

After the splitting is performed, we calculate the dissimilarities using (6. 8). Then RF dissimilarity is used as input of multi-dimensional scaling (MDS), which yields a set of points in the Euclidean space such that the Euclidean distances between these points are approximately equal to the dissimilarities. Finally, inspired by (Shi and Horvath 2006) we perform clustering using the output of the multi-dimensional scaling. To this end, we utilise the K-Means clustering algorithm (Lloyd 1982).

In the following we will discuss the construction of affinity matrix and discuss in detail the splitting criterion we employ for the decision tree construction.

#### **6.4.3.1. Affinity matrix**

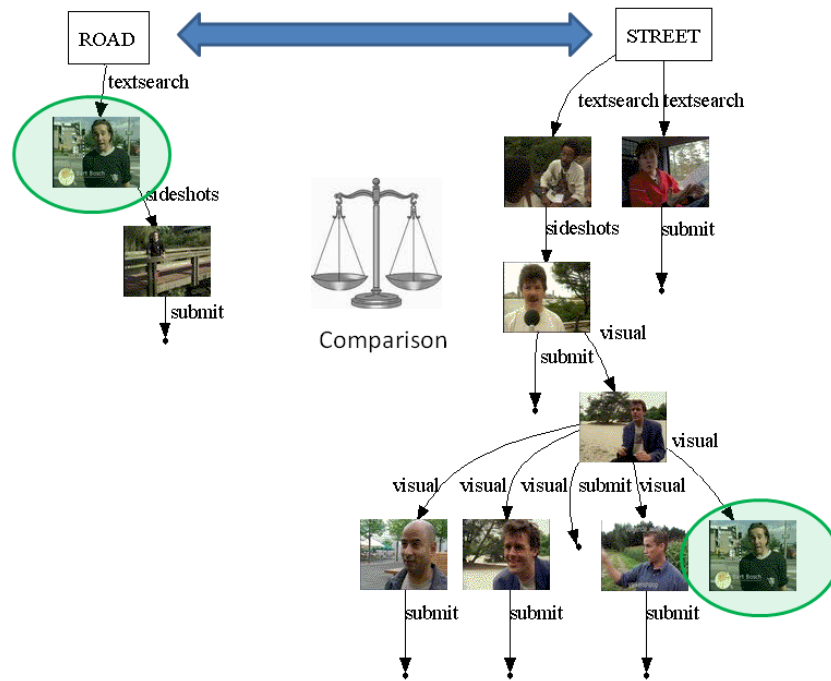
In the previous clustering method (section 6.3.2) we have generated an affinity matrix by considering the WordNet distances of the autonomous textual queries and the temporal information. As we have already discussed, in this scenario we do not consider the temporal information. Although the WordNet distances can give an indication of relevance, there are several cases, in which this metric fails to provide an acceptable result. Especially, when the context of the query is unknown (as in our case), the inability of term disambiguation (e.g. distinguish “jaguar” car and animal) further complicates the problem. To this end we propose to enrich this distance with semantic similarity on the involved images clicked during each subsession and which comprise the dependent queries.

Let’s define the semantic similarity between two subsessions  $A$  and  $B$ . The idea of this comparison is illustrated in Figure 6.6. As it was described in Chapter 4, each subsession includes one autonomous query and a set of dependent queries. We calculate the semantic similarity between the two autonomous queries using the WordNet similarity as described in section 6.3.2.1. However, the dependent queries consider keyframes (i.e. images) as input and therefore each subsession includes a set of images that were clicked by the user. To calculate a distance

between two sets of images we need to consider a metric that represents such a similarity.

One of the most well known metrics for set comparison is the Jaccard coefficient (Jaccard 1908). This coefficient measures similarity between sample sets, and is defined as the size of the intersection divided by the size of the union of the sample sets. Formally, for two sets  $A$  and  $B$ , the Jaccard similarity coefficient  $J(A, B)$  is given by:

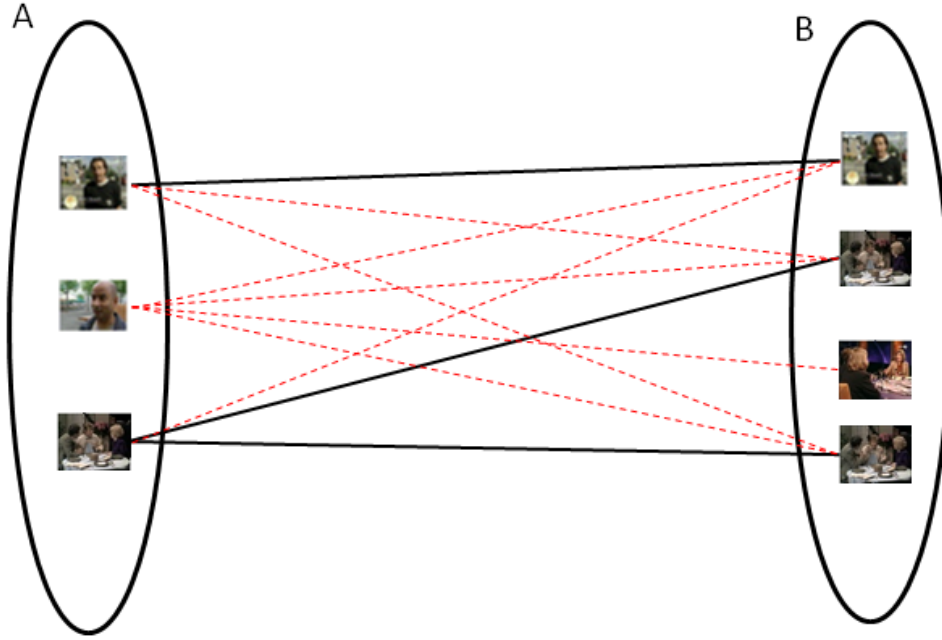
$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \quad (6.9)$$



**Figure 6.6. Comparison of subsessions. On the top the direct comparison between the textual queries is illustrated. The common images identified in the dependent queries are shown in green circles.**

However, when a set is comprised of images there are cases, in which several images are very similar to each other and can be considered near duplicates. In this context we proposed to enhance the Jaccard similarity coefficient and introduce the *visually enhanced Jaccard similarity*, which takes into account near

duplicates. We avoid introducing a similarity coefficient totally dependent on visual similarity due to the fact that this could lead to misleading results, since in many cases the visual distance doesn't correspond to the semantic distance.



**Figure 6.7. Sets A and B in a bipartite graph representation. Distances for non duplicate shots are represented with red dashed edges, while black solid edges indicate distances between near duplicates**

The idea is to identify near duplicate images between the different sets and consider them identical in order to compute the Jaccard similarity. However, the problem is not that simple, since each image might have more than one near duplicates and a random selection would lead to different results. For instance let us assume set  $A = \{a, b\}$  and  $B = \{c, d\}$ , where  $a$  is near duplicate with  $c$  and  $d$ , while  $b$  is near duplicate only with  $d$ . A random assignment would result to different similarity coefficients depending on the sequence we consider. In this case one assignment could be  $a \equiv c$  and  $b \equiv d$ , which leads to Jaccard similarity equal to 1, while another assignment would be only  $a \equiv d$  ( $b \equiv d$  cannot be considered, since each image is allowed to have only one near duplicate), which leads to Jaccard similarity equal to 0.5. It is obvious that from such a case the



most meaningful result is the first, while in the second case important parts of information are neglected.

Since the members of  $A$  are linked only with the ones of  $B$  (i.e. and no connections exist between them), we can represent these connections by considering a bipartite graph as it is shown in Figure 6.7.

Then we model the problem of identifying the maximum number of duplicates as a minimum weight perfect matching problem (or assignment problem) (Burkard, et al. 2009) in a bipartite graph. The minimum cost (weight) perfect matching problem is often described by the following story: There are  $n$  tasks to be processed on  $m$  agents and one would like to process exactly one job per machine such that the total cost of processing the jobs is minimised.

To this end we assign in each edge a cost  $c = 0$ , when the interconnected vertices represent duplicate images and  $c = 1$  when the images are not considered duplicates. This is performed by considering a distance threshold  $T$  as shown below:

$$c_{i,j} = \begin{cases} 0 & \text{if } c_{i,j} \leq T \\ 1 & \text{if } c_{i,j} > T \end{cases} \quad (6.10)$$

Then the problem is considered as a minimum weight matching, in which we want to identify a matching  $M$ , which minimises  $c$ . In this case the problem we face is non linear, since the member cardinalities of both sets are not necessarily equal. However, and given the fact that we are only interested in the assignments that have to do with the duplicate shots, we can easily transform it to a linear problem either by removing shots that do not have any near duplicate (i.e. remove shot  $i$  for which  $c_{i,j} = 1 \forall j$ ) or by introducing dummy shots that satisfy this requirement.

In order to solve this problem we apply the Hungarian algorithm (Kuhn 1955). Let assume a matching  $M$  between the shots of sets  $A$  and  $B$ . Its incidence vector would be  $x$  where  $x_{i,j} = 1$  if  $(i,j)$  belongs to  $M$  and 0 otherwise. Then the minimum weight perfect matching problem can be formulated as follows:

$$\text{Min} \sum_{i,j} c_{i,j} x_{i,j} \quad (6.11)$$

subject to:

$$\sum_j x_{i,j} = 1 \quad \forall i \in A$$

$$\sum_i x_{i,j} = 1 \quad \forall j \in B$$

$$x_{i,j} \geq 0, x_{i,j} \in \mathbb{N}, i \in A, j \in B$$

Then the Hungarian algorithm solves this problem in two steps: a) it constructs a cost matrix  $C_{n \times n}$ , where  $c_{i,j}$  is the cost for duplicating shot  $i$  and  $j$  and b) it uses equivalent matrix reduction to obtain the optimal assignment with respect to the cost matrix (Kuhn 1955).

**Table 6.2. Visually enhanced Jaccard similarity algorithm**

Input: the two image sets  $A = \{a_j\}$ ,  $B = \{b_i\}$

1. Eliminate any duplicate images separately in  $A$  and  $B$
2. Calculate all the visual distances  $d_{i,j}$
3. Transform the problem to a linear one by removing or introducing dummy shots.
4. Apply the Hungarian Algorithm to identify the best matching
5. Update the two sets  $A$  and  $B$  to  $\hat{A}$  and  $\hat{B}$  respectively after the identification of near duplicates (i.e. in case  $i$  and  $j$  are duplicates replace all  $j$  with  $i$ ).
6. Calculate the Jaccard similarity of the two updated sets  $\hat{A}$  and  $\hat{B}$

$$\text{Output} = eJ(A, B) = \frac{|\hat{A} \cap \hat{B}|}{|\hat{A} \cup \hat{B}|}$$

It should be noted that the Hungarian algorithm could be also used in order to solve the problem in the case that non-binary costs have been defined. An alternative could be to define as cost the distance between near duplicates and make the cost infinite between the non-near duplicate shots.

Finally, we propose to compute the Jaccard similarity after we have identified the maximum number of assignments between the images of the different sets based on near duplicates. The overall algorithm for calculating the enhanced Jaccard similarity is presented in Table 6.2.

Assuming that the WordNet similarity between the terms of the textual query is  $v_{i,j}$  as described in section 6.3.2.1 and  $eJ_{i,j}$  is the visually enhanced Jaccard similarity, the final similarity  $w_{i,j}$  is defined as:

$$w_{i,j} = \begin{cases} v_{i,j} \cdot eJ_{i,j} & \text{where } v_{i,j}, eJ_{i,j} \neq 0 \\ v_{i,j}, & \text{if } eJ_{i,j} = 0 \\ eJ_{i,j}, & \text{if } v_{i,j} = 0 \end{cases} \quad (6.12)$$

#### 6.4.3.2. Splitting criterion for decision tree construction

In this section we define formally the splitting criterion, which we apply during the decision tree construction of the gaze-driven random forests in order to identify the best split (Figure 6.6).

We select the Gini Index, which is also used by Breiman for RF construction (Breiman, et al. 1984) as the basis of our impurity function:

$$G = \sum_{j=1}^K p_j(1 - p_j) = 1 - \sum_{j=1}^K (p_j)^2 \quad (6.13)$$

In order to incorporate the homogeneity of the samples that are clustered after the split, we introduce a new variable called the *homogeneity co-efficient*. This represents the homogeneity of a set of samples considering the user interest reflected by the aggregated gaze movements. It should be clarified that, while the Gini index is based on  $p_j$ , which is the probability of a sample belonging to the observed or the synthetic class, the homogeneity co-efficient depends on the  $p'_i$ , which corresponds to the probability that a query belongs to topic  $i$ .

We calculate the homogeneity co-efficient by employing the gaze trained SVM model. Let's assume that we have  $M$  points in node  $t$ . It should be noted that

these would include  $K < M$  queries and  $L = M - K$  vectors that belong to the synthetic data. For the queries that fall into the same split, we assume that they belong to the same cluster (topic) and we aggregate the gaze features for the  $S$  shots that resulted from these queries and for which, fixations have been identified. The output of the classifier provides as result the distance  $d_i$  between each shot  $i$  and the hyperplane. Then, the homogeneity co-efficient is calculated by considering these distances in a sigmoid function:

$$h = \frac{1}{1 - e^{-\frac{\sum_{i=1}^S |d_i|}{S}}} \quad (6.14)$$

We incorporate the homogeneity coefficient in the impurity function of (6.13):

$$i(t) = \frac{1}{h} (1 - \sum_{j=1}^K (p_j)^2) \quad (6.15)$$

Given the fact that  $h$  is based on a sigmoid function it ranges in  $[0, 1]$ . Finally, based on (6.6), (6.14) and (6.15) the splitting criterion, which we need to maximise, is:

$$\Delta i(s, t) = \frac{1}{h} (1 - \sum_{j=1}^K (p_{Tj})^2) - \frac{w_R}{h_R} (1 - \sum_{j=1}^K (p_{TRj})^2) - \frac{w_L}{h_L} (1 - \sum_{j=1}^K (p_{TLj})^2) \quad (6.16)$$

Given the fact that  $K = 2$ , since only two classes are considered (i.e. synthetic and observed data) the splitting criterion  $\Delta i(s, t)$  becomes:

$$\frac{1}{h} (1 - p_{T1}^2 - p_{T2}^2) - \frac{w_R}{h_R} (1 - p_{TR1}^2 - p_{TR2}^2) - \frac{w_L}{h_L} (1 - p_{TL1}^2 - p_{TL2}^2) \quad (6.17)$$

The  $h_R$  is the homogeneity co-efficient, which corresponds to the probability that the samples in the right split of node  $T$  belong to the observed or synthetic data and measures the purity of the observations in a node. In a similar way the  $h_L$  is the homogeneity co-efficient for the left split, while  $h$  represents the homogeneity co-efficient before the split. As  $p_{Tj}$ ,  $p_{TRj}$  and  $p_{TLj}$  the declared the

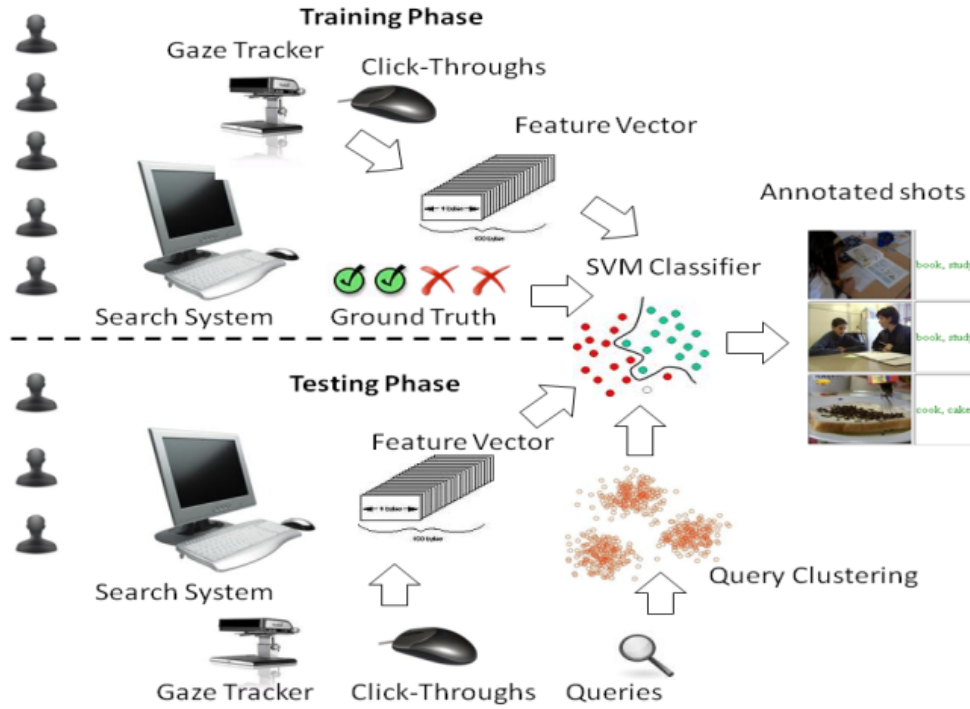
probabilities of the observations in node  $T$  belonging to class  $j$  before the split, and in the right and left split respectively.

It should be mentioned that the aggregation plays a very important role. If the features have not been aggregated, this approach could fail in the case that interesting shots for different topics would have been sent to the classifier. However, due to the feature aggregation, shots that have received user attention during different topics would result to non descriptive features, which would make the classifier underperform. Therefore the performance of the classifier can be considered as an additional indicator of how homogeneous a set of shots and respectively a set of queries can be considered.

## 6.5. Results and evaluation

In order to evaluate the proposed framework, the clustering algorithms and the produced annotations, we have used the user data gathered during the video retrieval experiment described in section 5.3.1. Then, we evaluate the two clustering algorithms, we generate annotations based on their performance and finally we compare them.

We recall that in this experiment 8 users were recruited to search for 4 different topics A-D using the LELANTUS interactive video search engine. During the search tasks the gaze movements, the mouse clicks and query submissions were recorded. However, in order to simulate the situation of users searching subsequently (as this is required by the scenario discussed in section 6.3), the timestamps of the search actions were synchronised in such a way so that the topic search sessions appear sequentially. In other words, the time breaks between two search topics were eliminated. It should be noted that the temporal information was important only for the dominant set clustering solution, while for the gaze-driven random forest clustering the temporal dimension was not considered. The experiment and the procedure for annotation, which realises the proposed framework, are depicted in the schematic view of Figure 6.8.



**Figure 6.8. Interactive experiment and video annotation**

As discussed in section 6.2, during the training phase, we consider known search topics and we exploit this information to generate gaze-based features in order to build the interest prediction classifiers, which are described in detail in Chapter 5. In the testing phase we assume that the topics are unknown and we represent them by clusters of queries. The query clusters are produced by considering the two proposed algorithms (i.e. temporal clustering and gaze-driven random forest clustering).

In the following, we present the methodology we employ to evaluate and compare the clustering methods. Then, we provide annotation results, when each of the proposed clustering methods is employed.

### 6.5.1. Clustering evaluation methodology

Over the past few decades, hundreds of clustering algorithms have been devised. With a view to evaluate and compare them, various clustering comparison measures have been proposed. The most popular include the class of pair-counting based measures such as the well-known Adjusted Rand Index (Hubert and Arabie 1985), and set-matching based measures, such as the  $H$  criterion

(Meila 2005), information theoretic based measures, such as the Mutual Information (Strehl and Ghosh 2002) and the Variation of Information (Meila 2005), form another fundamental class of clustering comparison measures. In this work we have selected to use the normalised version of the Mutual Information metric  $NMI$ , which according to (Vinh et al. 2009), is preferable in many applications. In the following, we present the normalised version of the Mutual Information metric  $NMI$ .

Assuming that we have a set of  $N$  data items and two clustering solutions  $U$  and  $V$  (e.g. the dominant or random forest clustering solution and the ground truth),  $U = \{U_1, U_2, \dots, U_R\}$  with  $R$  clusters and  $V = \{V_1, V_2, \dots, V_C\}$  with  $C$  clusters, the  $NMI$  is defined as:

$$NMI(U, V) = \frac{I(U, V)}{\sqrt{H(U)H(V)}} \quad (6.18)$$

In (6.18)  $H(U)$  is the information entropy of a clustering solution  $U$ :

$$H(U) = - \sum_{i=1}^R P(i) \log P(i) \quad (6.19)$$

where  $P(i) = \frac{|U_i|}{N}$ . Similarly the  $H(V)$  is calculated as:

$$H(V) = - \sum_{j=1}^C K(j) \log K(j) \quad (6.20)$$

where  $K(j) = \frac{|V_j|}{N}$ . Finally the mutual information between these two clustering solutions is calculated as:

$$I(U, V) = \sum_{i=1}^R \sum_{j=1}^C P(i, j) \log \frac{P(i, j)}{P(i)K(j)} \quad (6.21)$$

where  $P(i, j)$  denotes the probability that a point belongs to cluster  $U_i$  in  $U$  and cluster  $V_j$  in  $V$ :  $P(i, j) = \frac{|U_i \cap V_j|}{N}$  (Strehl and Ghosh 2002).

### 6.5.2. Training and testing

In this evaluation, we have considered as training data the search sessions performed by 5 users. The results submitted by these users constitute an explicit relevance metric with respect to the query topics for all the viewed items. In order to evaluate the framework, we consider several cases, in which different clustering algorithms, combinations of topics and users are used for training and testing. More specifically, the following cases, which are shown in Table 6.3 are considered: in the first case, we train recursively 6 different classifiers (models 1-6 in Table 6.3) by selecting each time a different combination of two topics (i.e. (A, B), (A, C), etc.) and using as vector the 1-5 fixation-based features (Table 5.1) and we consider the topics known during testing. In the second case (models 7-12 in Table 6.3) we repeat the same scenario but considering the test query topics as unknown and by employing the dominant set clustering algorithm. The purpose of these two first variations is to investigate the performance of the system when the clustering approach is applied (i.e. for unknown search topics). For cross-validation purposes we average the results from 6 topic variation and from 2 different user train and test variation. Specifically in the first scenario we consider the data of the users 1-5, while at the other we use the data of the users 4-8. In the third case, we employ the gaze-driven RF clustering approach and we consider the same topics for training and testing. In order to evaluate the clustering algorithm during testing we average the results from all the possible 5-3 combinations of training and testing users, which leads to 56 user variations. Finally in the forth case we employ the gaze-driven RF and we consider all the different combination of two topics, as well as the same 2 user variations as in case 1 and 2. In all cases, grid search is employed to select the best SVM training parameters.

**Table 6.3. Training and testing cases**

Case	Model No	Training-Testing topics	Users	Clustering	Merge
1	1-6	2-2	2 variations	Initial Topics	Topics
2	7-12	2-2	2 variations	Dominant Sets	Clusters
3	13	4 same	56 variations	Gaze driven RF	Clusters
4	14-19	2-2	2 variations	Gaze driven RF	Clusters



### 6.5.1. Results of topic-based merging

In this section, we evaluate the proposed framework by reporting the classification accuracy, the precision, the recall, the average precision (AP) and the F-score over the items returned by the system as positive results. Although similar results (for different topic variations) are presented in Chapter 5, we report them for comparison purposes. Specifically, we use these results as a baseline in order to investigate how much the classification performance drops when query clusters (which are expected to include noise) instead of the initial topics are considered.

During testing the submitted results by the 3 test users formed the golden set for the evaluation. Formally, assuming that the classifier returns  $TP$  true positives,  $TN$  true negatives,  $FP$  false positives and  $FN$  false negatives for a topic calculated against the  $V$  positive and the  $N$  negative user selections, the accuracy is computed as  $A = \frac{TP+TN}{V+N}$ , the precision as:  $P = \frac{TP}{TP+FP}$ , the recall as:  $R = \frac{TP}{V}$ . We mostly judge the performance of the system using F-Score, due to the fact that the considered data are imbalanced (i.e. very few positive examples compared to negatives) and therefore judging only by the accuracy could be misleading (e.g. marking all the results as negative could provide an accuracy of 90%). The results for the aforementioned training cases using train data from two different user variations are reported in Table 6.4.

**Table 6.4. First case (Topic-based merging)**

Model	Train Topics	Test Topics	Clas. Acc.	Precision	Recall	AP	F-Score
1	A,B	C,D	96.91%	69.8%	38.2%	72.9%	49.38%
2	A,C	B,D	95.83%	71.97%	45.22%	73.2%	55.54%
3	A,D	B,C	95.44%	66.33%	41.12%	69.6%	50.76%
4	B,C	A,D	96.9%	52.12%	66.7%	68.3%	58.51%
5	B,D	A,C	96.5%	43.3%	69.7%	66.5%	53.42%
6	C,D	A,B	96.83%	67.12%	52.12%	72.8%	58.67%
<b>Average</b>			<b>96.40%</b>	<b>61.77%</b>	<b>52.17%</b>	<b>70.55%</b>	<b>54.38%</b>

## 6.5.2. Results of dominant set clustering

### 6.5.2.1. Evaluation of clustering

The application of dominant set clustering is applied in the second case. In Table 6.5 we can see the average *NMI* calculated for the two different user data variations. The average *NMI* is calculated as 0.48, while in average 8 unique queries are grouped together during this phase generating 10.25 clusters. It is interesting to notice that in all cases the generated clusters exceed the number of the initial topics. However, as we will discuss later, this doesn't introduce necessarily an error.

**Table 6.5. Second case (Cluster-based merging).**

Train Topics	Test Topics	Clusters	Unique queries	NMI
A,B	C,D	9	75	0.465
A,C	B,D	16	99	0.493
A,D	B,C	6	77	0.455
B,C	A,D	15.5	79.5	0.45
B,D	A,C	9.5	63.5	0.55
C,D	A,B	10.5	98	0.465
<b>Average</b>		<b>10.25</b>	<b>82</b>	<b>0.48</b>

### 6.5.2.2. Classification evaluation







Although the ground truth in the topic-based merging is straightforward, since we have the explicit result submissions by the users during the retrieval tasks, in cluster-based merging it is not that clear. To evaluate the classification performance we make the assumption that the queries are clustered acceptably. As acceptable cluster we consider any clustering solution, in which the queries that are labelled with one cluster do not belong to different topics and therefore it is reasonable to associate them with the initial topics. However, it is true that this assumption is not always valid as it depends on the clustering performance and for this reason we provide a separate evaluation and discussion in the next subsection.

The Classification accuracy, the precision, recall, average precision and the F-Score for the two user variations are presented in Table 6.6. To rank the classifier results we take into account the distance from the hyperplane, which

discriminates the two different classes. The ranking is evaluated by reporting the AP, which reaches 69.57%, while the precision is 62.29%, which shows that the correct annotations produced by the system could be further improved by introducing thresholds or increasing the precision at the cost of reducing the recall.

**Table 6.6. Second case (Cluster-based merging)**

Model	Train Topics	Test Topics	Clas. Acc.	Prec.	Rec.	AP	F-Score
7	A,B	C,D	95.81%	70.29%	37.1%	70%	48.57%
8	A,C	B,D	95.96%	67.64%	40.42%	71.9%	50.6%
9	A,D	B,C	94.29%	66.32%	36.82%	65.51%	47.35%
10	B,C	A,D	96.81%	58.08%	65.67%	68.9%	61.64%
11	B,D	A,C	96.12%	44.11%	62.77%	69.8%	51.81%
12	C,D	A,B	95.9%	67.31%	45.82%	71.33%	54.52%
<b>Average</b>			<b>95.81%</b>	<b>62.29%</b>	<b>48.1%</b>	<b>69.57%</b>	<b>52.42%</b>

Shot	Annotation	Score	Topic	Shot	Annotation	Score	Topic
	book, study	5.676045	C		book, study	4.696814	C
	cook, cake	4.560157	D		book, study	4.128421	C
	cook, cake	3.708860	D		cook, cake	3.143348	D

**Figure 6.9. Automatic annotations using model 7**

### 6.5.2.3. Annotated shots

The quality of the annotations strongly depends on the two aforementioned evaluation dimensions (i.e. clustering and classification). An ideal query clustering will actually lead us to the initial topic-based merging and the performance will depend only on the classifier. On the other hand, a low quality

clustering could result to low performance, since queries of different topics will be in the same cluster and annotated wrongly. However, the actual problem occurs only when queries of different topics are associated with the same cluster label and not when one topic is divided into more clusters.

To this end we provide an explicit judgement of the produced annotations and compare with the previous evaluations. In Table 6.7 we see how the initial precision of the system drops due to clustering errors from 62.3% to 51.6%. In addition we report that around 120 shots are annotated during a full session of 3 users searching for 2 topics (i.e. 1 hour in total).

**Table 6.7. Produced annotations for second cases and training data from users 1-5 and 4-8**

Model	Class. Prec.	F-score	NMI	Correct annotations	Anno. shots	Final Precision
7	70.29%	48.57%	0.465	57	107	53.27%
8	67.64%	50.6%	0.493	87	129	67.44%
9	66.32%	47.35%	0.455	49	118	41.52%
10	58.08%	61.64%	0.45	77	130	59.23%
11	44.11%	51.81%	0.55	70	156	44.8%
12	67.31%	54.52%	0.465	47	104	45.19%
<b>Average</b>	<b>62.3%</b>	<b>52.4%</b>	<b>0.48</b>	<b>64.5</b>	<b>124</b>	<b>51.6%</b>

A visual example of the annotations provided by model 7 (Table 6.7) is shown in Figure 6.9. In this case the shots are annotated with the two most frequent words describing each cluster. For instance, the topic C “Find shots of one or more people with one or more books” was labelled with “book”, “study”, while the topic D “Find shots of food and/or drinks on a table”, was labelled as “cook”, “cake”.

### 6.5.3. Results and evaluation for gaze-driven random forest clustering

In order to evaluate this approach we compare the gaze-driven random forest clustering (G-RF) with a baseline such as K-means, as well as with the traditional random forest clustering (RF) (i.e. without the employment of gaze information).

First, in order to evaluate the performance of the clustering algorithm we consider the third case, in which all the 4 topics are involved. For cross-validation purposes, we perform the clustering for all the 56 possible user variations, in which the data of the five users are used for training and the data of the remaining three for testing. In average the training query submissions are 396.6, while the testing queries have been 238.2.

An ideal clustering method would group these queries into 4 different clusters. Given the fact that we consider the topics unknown (in terms of cardinality and subject) we attempt to cluster the queries in different number of clusters and evaluate the performance of different algorithms using the *NMI* metric.

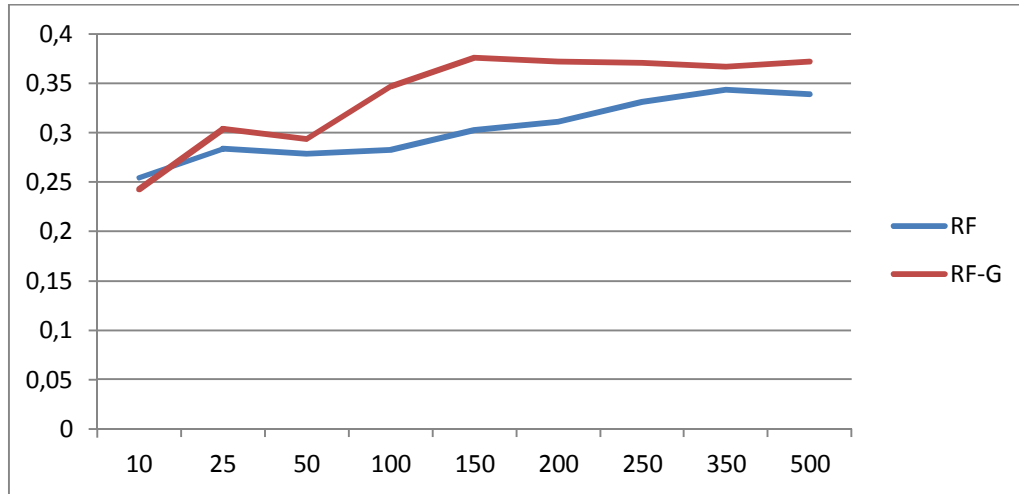
For each random forest we have constructed a variety of trees ranging from 10 to 500, and we assume 15 random variables. The number of random variables is selected to be close to the square root of the feature number (i.e. 238.2 in average for all the user variations). We have decided to stop at 500 trees, since we observe that the results have started to converge. After tuning experimentally the parameters for  $h$  (6. 14) we have selected  $a = 40$  and  $b = 40$ .

In the following, we present the average cross-validated results for all the 56 user variations, when different cardinality of clusters is considered. At the same time we compare the results of RF without considering the gaze movements.

In Table 6.8 we report the results for 4 clusters. In this case, the *NMI* of the gaze-driven RF outperforms the traditional RF method by an average of 8.9%. A graphical view of the results is available in Figure 6.10.

**Table 6.8. NMI for the gaze-driven RF for 4 clusters**

clusters =4									
trees	10	25	50	100	150	200	250	350	500
RF	0.25	0.28	0.28	0.28	0.30	0.31	0.33	0.34	0.34
RF-G	0.24	0.3	0.29	0.35	0.376	0.37	0.371	0.367	0.37

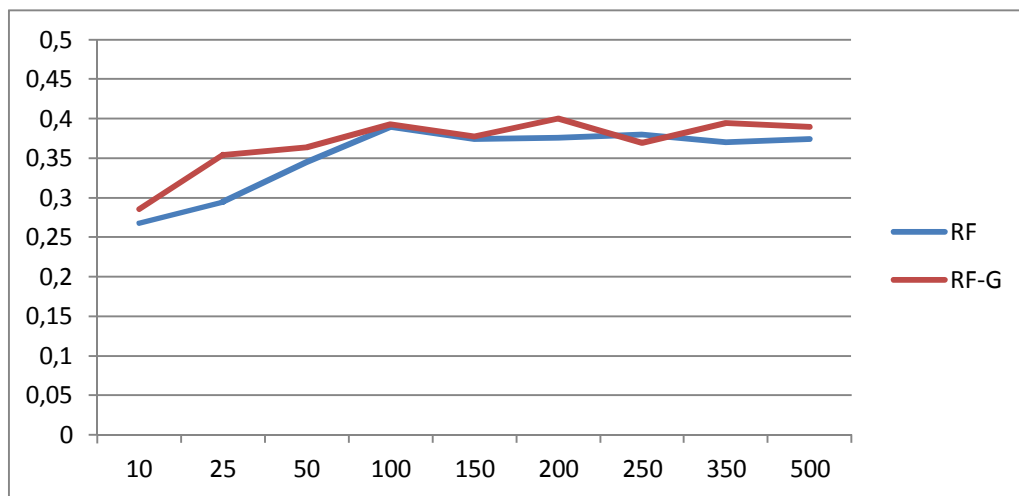


**Figure 6.10. The gaze-driven RF NMI performance for 4 clusters**

When the number of clusters is increased to 6, the results (Table 6.9) seem to be improved reaching a final *NMI* of 0.39. However, the increase compared to the traditional RF is lower since it reaches 5.4%. A visual view of these results is provided in Figure 6.11, in which we observe that the performance of both algorithms is very close and only seems to differentiate when we reach the 500 trees.

**Table 6.9. NMI for the gaze-driven RF for 6 clusters**

clusters =6									
trees	10	25	50	100	150	200	250	350	500
RF	0.267	0.294	0.345	0.389	0.374	0.376	0.38	0.37	0.37
RF-G	0.285	0.35	0.364	0.393	0.377	0.4	0.369	0.39	0.39

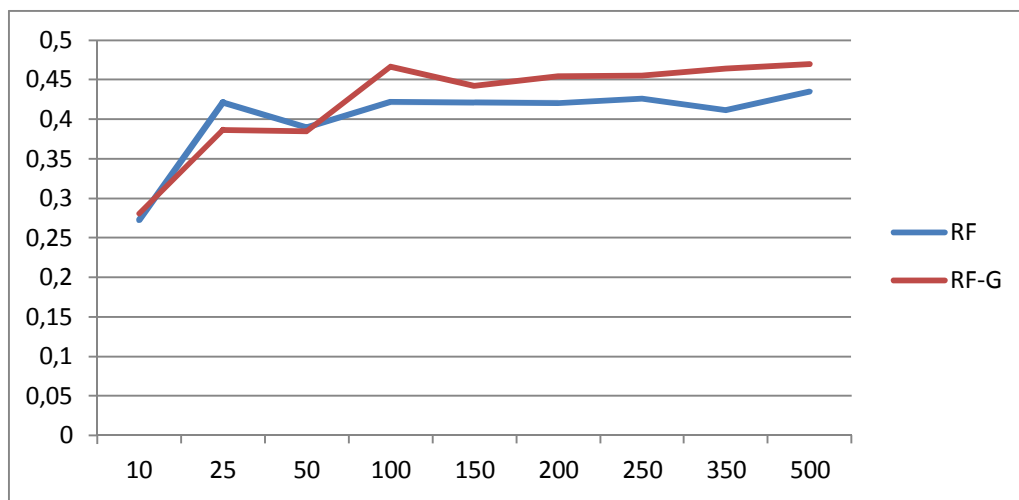


**Figure 6.11. The gaze-driven RF NMI performance for 6 clusters**

In Table 6.10 we present the results for 8 clusters. It is clear that the algorithm performance has been increased in comparison with the 4 and 6 clusters. In this case the results of gaze-driven RF outperform RF with an average of 8%. A visual view of the results is provided in Figure 6.12.

**Table 6.10. NMI for the gaze-driven RF for 8 clusters**

clusters =8									
trees	10	25	50	100	150	200	250	350	500
RF	0.27	0.425	0.389	0.421	0.421	0.42	0.426	0.411	0.435
RF-G	0.28	0.38	0.384	0.467	0.4425	0.455	0.455	0.464	0.47

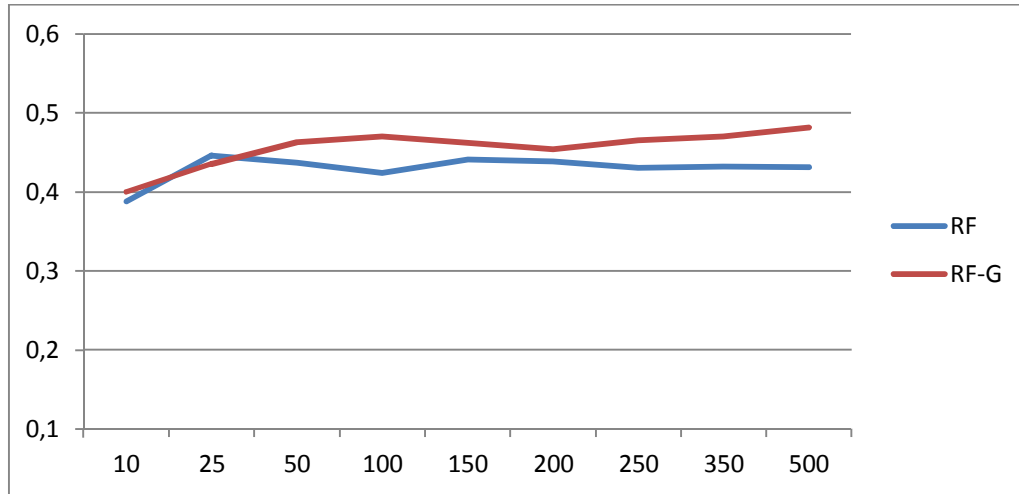


**Figure 6.12. The gaze-driven RF NMI performance for 8 clusters**

Finally we report the results in the case of 10 clusters. In this case the gaze-driven RF demonstrate a performance of 0.48, which improves the performance of traditional RF by a 11.8%. The better performance of gaze-driven RF is also illustrated in Figure 6.13.

**Table 6.11. NMI for the gaze-driven RF for 10 clusters**

clusters =10									
trees	10	25	50	100	150	200	250	350	500
RF	0.39	0.45	0.437	0.424	0.44	0.4384	0.43	0.43	0.43
RF-G	0.4	0.44	0.463	0.47	0.462	0.4541	0.466	0.47	0.481



**Figure 6.13. The gaze-driven RF NMI performance for 10 clusters**

It should be mentioned that the fluctuation that is observed in the low number of trees (e.g. 10, 25) is reasonable due to the high randomness introduced in this case. When the number of trees is increasing the algorithm seems to converge and not big fluctuations are reported.

Finally, in Table 6.12 we report the results for the 2 aforementioned techniques for 500 trees and compare them with the K-means baseline. A graphical comparison of these results is available in Figure 6.14. First it is interesting to notice that *NMI* is in general increasing together with the number of clusters. This is probably due to the fact that in this way we avoid associating queries with totally irrelevant clusters. In average the gaze-driven RF (G-RF) performs better. Finally, in Figure 6.15 we present the average performance of the three clustering algorithms along all the clusters.

**Table 6.12. NMI comparison for the 3 clustering techniques for 500 trees**

Num of clusters/ Technique	K-means	RF	G-RF
4 clusters	0.2834	0.339	0.372
6 clusters	0.3534	0.374	0.39
8 clusters	0.3677	0.435	0.47
10 clusters	0.3734	0.432	0.48
Average	0.344	0.3949	0.4283





Figure 6.14. The performance for the 3 clustering algorithms

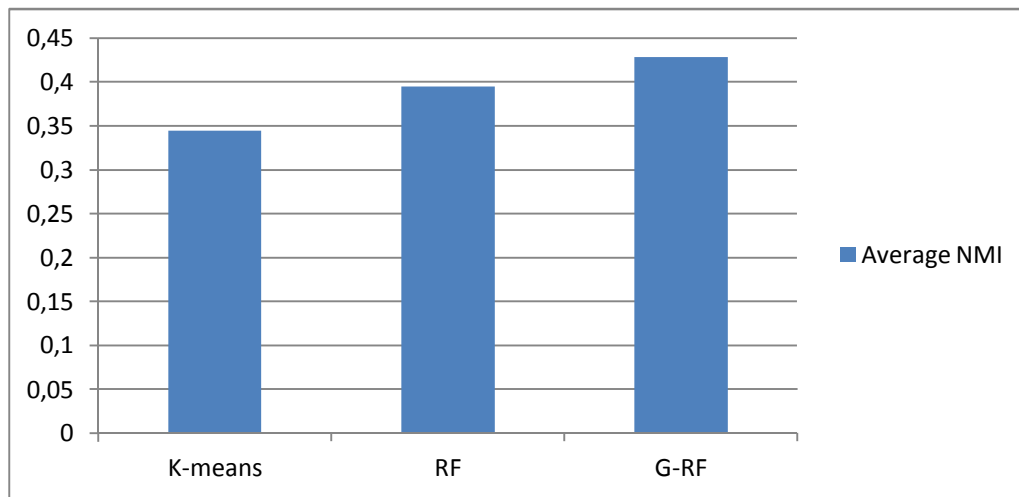


Figure 6.15. The average performance for the 3 clustering algorithms

In general it seems that the gaze-driven RF outperforms RF and K-means by about 24.5% and 8.45% respectively.

Table 6.13. Classification performance of the forth case (gaze-driven RF cluster-based merging)

Model	Train Top.	Test Top.	Clas. Acc.	Prec.	Rec.	AP	F-Score
14	A,B	C,D	95.05%	64.46%	41.71%	83.1%	50.6%
15	A,C	B,D	95.33%	70.49%	50.59%	79.0%	58.9%
16	A,D	B,C	95.07%	65%	43.62%	75.6%	52.2%
17	B,C	A,D	96.67%	58.77%	57.27%	76.1%	58%
18	B,D	A,C	96.05%	52.94%	52.07%	56.9%	52.5%
19	C,D	A,B	95.70%	51.22%	60.58%	69.8%	55.5%
<b>Average</b>			<b>95.65%</b>	<b>60.48%</b>	<b>50.97%</b>	<b>73.4%</b>	<b>54.6%</b>

### 6.5.3.1. Classification performance

The previous section was dedicated in the evaluation of the gaze-driven clustering algorithm performance. As we have discussed, the final annotations strongly depend both on the clustering and the classification performance. In Table 6.13 we report the classification results for the forth case.

### 6.5.3.2. Annotated shots

Finally, we employ the gaze-driven RF clustering algorithm to generate annotations results considering the forth case. After considering two different user variations we report the annotations in Table 6.14. In this case more than 116 shots were annotated in average. It is also interesting to notice how the initial precision is decreasing from 60.46% to 55.57% due to the error introduced by the clustering algorithm.

**Table 6.14. Annotations of the forth case (gaze-driven RF cluster-based merging)**

Model	Class. Prec.	F-score	NMI	Correctly annotated	Annotated shots	Final Precision
14	64.4%	50.65%	0.51	52	121	43%
15	70.5%	58.90%	0.58	81	122	66.3%
16	65%	52.21%	0.51	62	100	62%
17	58.8%	58.01%	0.57	68	114	59.6%
18	52.9%	52.5%	0.54	61	119	51.26%
19	51.2%	55.51%	0.5	59	123	47.9%
<b>Average</b>	<b>60.46%</b>	<b>54.63%</b>	<b>0.535</b>	<b>63.83</b>	<b>116.5</b>	<b>55.57%</b>

### 6.5.4. Comparison of annotations

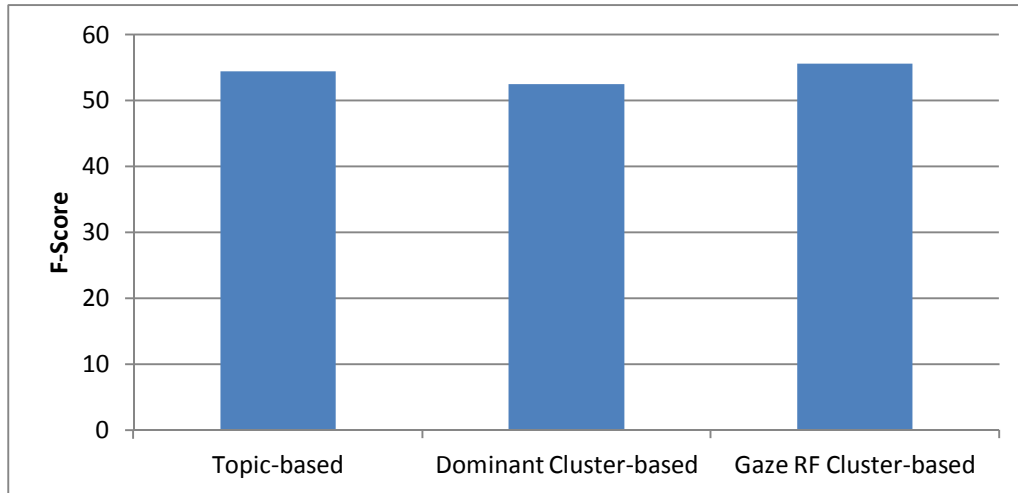
Finally, in this section we compare the performance of the two clustering algorithms and the produced annotations, when the same user variations are employed (cases 2 and 4).

In Table 6.14 we report the comparison of the classification module between the topic-based merging, the dominant set and the gaze-driven RF cluster-based merging. Despite that we perform a cluster-based merging in the last two cases the performance of the system is almost stable. Specifically the F-score reports an average decrease of 4.6%, when the dominant set algorithm is employed,

while it is slightly increased by a 2.2% when the gaze-driven RF is applied. This is also illustrated more clearly in Figure 6.16.

**Table 6.15. Results for different training user data variations**

Merging	Topics	Precis.	Recall	AP	F-Score
Topic-based	2-2	61.8%	52.2%	70.6%	54.38%
Dominant set Cluster-based	2-2	62.3%	48.1%	69.6%	52.42%
Gaze RF Cluster-based	2-2	60.47%	50.97%	73.4%	55.57%



**Figure 6.16. F-Score performance for different merging approaches**

Finally, in Table 6.16 we present a direct comparison of the annotations for the aforementioned cases. It is interesting to notice that for the same exactly user variations and therefore submitted queries, the gaze-driven RF demonstrates a better F-score by an average of 6.67%. Although more annotations are produced by dominant set clustering, the quality is better when the gaze-driven RF algorithm is employed. As it is illustrated in Figure 6.17 the initial classification precision drops around 17% and 13.3% for the cases of dominant set and gaze-driven RF clustering respectively.

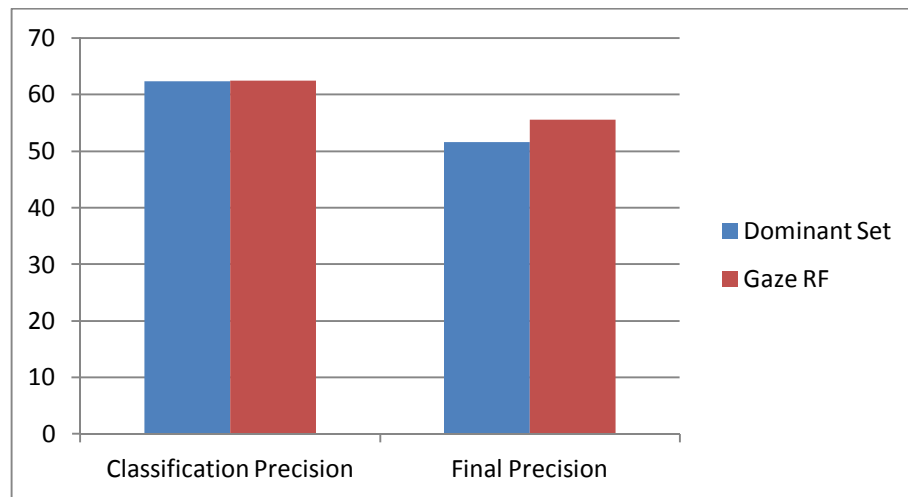
**Table 6.16. Average produced annotations for second and forth training cases and training data from users 1-5 and 4-8**

Clustering method	Class. Prec.	F-score	NMI	Correct annotations	Annotated shots	Final Precision
Dominant set	62.3%	52.4%	0.48	64.5	124	51.6%
RF-G	62.4%	54.63%	0.535	63.83	116.5	55.04%

Although we have presented a comparison between these two annotation approaches, it should be mentioned that the clustering algorithms involved could not be directly compared due to the fact that they consider different scenarios and input data. Specifically the dominant set clustering method takes into account the temporal information, which is not exploited in the gaze-driven RF. On the other hand the RF-based approach performs subsession (instead of autonomous query) clustering and takes into account also the gaze movements for the generation of the clusters.

## 6.6. Conclusions

This chapter describes a video annotation framework that combines supervised and unsupervised methods that exploit the gaze movements and clicks during video retrieval. In this context we have proposed two different clustering algorithms (dominant set and gaze-driven random forests). The first is based on dominant set algorithm and considers textual similarity between queries and the temporal dimension. The gaze-driven RF is a more sophisticated algorithm which relies upon textual and visual similarity between the queries and it is driven by the performance of the shot gaze-based classification.



**Figure 6.17. Classification and final precision for different clustering methods.**

The results show that the algorithm of gaze-driven RF usually outperforms the dominant sets despite the fact that temporal information is not taken into account. Although the results are based in an experiment with limited number of users and

topics, they can be considered as an important indication that such techniques could be used effectively for video tagging and annotation purposes.

## **Chapter 7**

### **CONCLUSIONS**

*This chapter summarises the achievements reported in this thesis and discusses future work and research challenges.*

#### **7.1. Summary of achievements**

In this thesis we have focused on exploiting implicit user feedback in the context of interactive video retrieval. We have dealt both with the implicit user feedback that is hidden under the user navigation patterns and gaze movements. To achieve our goals we have designed and implemented an interactive video retrieval framework, which indexes the video with the aid of content-based analysis, as well as with analysis of the past user interaction data and gaze movements.

As far as past user navigation patterns are concerned, a novel methodology of past user interaction modelling was proposed, which is based on query categorisation and subsession generation. By aggregating past user interaction we construct a graph that reflects the user navigation patterns during a video retrieval task. This graph is utilised to provide recommendations to future users by considering the video shot distances in the graph. In addition, a further optimisation of the results provided by content-based retrieval modalities (i.e. visual search) is achieved with the aid of a SVM classifier, which is trained with the aid of graph structured user aggregated interaction data. As it is shown by the results and the evaluation, the past user data can be of added value in modern video retrieval engines as large amounts of user implicit feedback are available especially in web applications.

Then, we have investigated the role of gaze movements during interactive video retrieval tasks. We have attempted to discriminate between relevant and non relevant shots to a given query topic with the aid of SVM classifiers and gaze-based features. Specifically, fixation and pupil dilation-based features generated from aggregated user eye movement data have been proposed to train the classifier. To evaluate the proposed methodology, experiments with an interactive video search engine are conducted. The results show that gaze-based implicit feedback could be of added value in interactive video retrieval tasks, since it can be considered as an important indicator regarding the relevance of a video shot to a query topic even in not strictly controlled environments.

Finally, we have proposed an automatic video annotation framework that combines unsupervised (clustering) and supervised (classification) machine learning approaches with a view to automatically annotating video content. In this context, we cluster the submitted queries in semantic topics and identify relevant shots based on aggregated gaze movements using the aforementioned classifier. Two query clustering techniques have been presented: a) dominant set clustering based on temporal and textual information, b) unsupervised random forests grown utilising gaze movements, textual and visual information. The results show that aggregated gaze movements can be exploited effectively for automatic video tagging and annotation purposes.

The significance of the achievements of this thesis is reflected by the research contributions in the video retrieval domain, as well as by the potential for the development of innovative applications in this field.

First, the research achievements and the techniques developed in this thesis contribute to the state of the art in interactive video retrieval by alleviating existing problems such as the semantic gap, the detection of user interest and automatic video annotation. Specifically, we proposed an efficient combination of content-based modalities with aggregated implicit user feedback provides an alternative way to bridge the semantic gap. In addition, the generation of gaze-based features for multimedia items provides the capability of judging an image/shot with respect to the user interest, while the gaze-driven clustering

allows for grouping semantically similar queries into clusters to support automatic annotation of video content.

Second, the proposed techniques have been developed in accordance with the requirements and trends imposed by the advancements of information technologies such as the need to search in large multimedia content and the usage of sensors. Especially, the constantly increasing usage of wearable sensors and kinetics to improve the user experience is an important factor that affects human-computer interaction. The proposed work takes into account these trends and considers interaction data from aggregated voluntarily (i.e. mouse strokes) and involuntarily user responses (i.e. gaze movements), in order to gain understanding on the multimedia content and improve video search.

Finally, the proposed developments could support the implementation of video search and retrieval both in web applications, as well as in personal digital collections (e.g. photos and videos), which are widely used by everyday users. The recently emerging social media platforms would also benefit from such technologies, since the user interaction data during content browsing and exchange could be also processed to facilitate search tasks.

## **7.2. Future work**

Although the results presented in this thesis are important indicators regarding the exploitation of implicit user feedback, there are still several open challenges, as well as future work that should be conducted. In this context we first discuss in detail the goals of the future work and then we present additional research challenges based on the proposed thesis.

First, it would be very interesting to conduct experiments with large number of users in different environments, in order to further investigate the performance of the proposed algorithms. In this context, future work includes the incorporation of the proposed techniques in a web multimedia search engine, in which many users are using in a daily basis. Such experiments could show the added value of the graph-based representation of past user interaction when dealing with large amounts of data and also reveal any scalability issues that have to be addressed.



In addition, more intelligent algorithms for recommendation generation based on the aforementioned graphs have to be investigated. Specifically, the algorithms have to consider not only the distance between two vertices but also the local density of the graph between the vertices of interest.

As far as the gaze movement investigation is concerned, future work includes the involvement of additional eye movement features to further enhance the proposed representation. Specifically, we propose to include scan paths and saccades in order to create a feature vector that represents in a more efficient way the user interest based on eye movements. Feature selection strategies could be employed in order to identify which features are more important for cognitive user behaviour and investigate whether the scanpath and saccade features can complement the fixation and pupil dilation information. In addition, complementing these features with click-based features has to be investigated. This approach should be compared to late fusion methods, in which the user navigation patterns and the gaze movements are modelled with different feature vectors and the fusion is performed at the decision level.

Finally, regarding the automatic annotation based on gaze movements, the future work includes additional experiments with more users and topics, as well as further optimisation of the method. Further optimisation of the algorithm includes the assessment of the classification result based on textual and visual features of the classified shots, different approaches for creating synthetic data for unsupervised random forests, as well as the incorporation of click, scan path and saccade features with early and late fusion as discussed above.

In the following, we present additional research activities that are triggered by the proposed thesis.

#### **7.2.1. On line video retrieval system based on implicit user feedback**

After having developed techniques to recommend video content utilising gaze movements and past user navigation patterns in an off line framework, a challenging objective would be to implement an on line retrieval system that generates recommendations considering also the real time feedback by the user, providing in that way a more context-oriented response.

To this end, the following approaches could be considered. A first idea to exploit the graph-based representation of past user interaction includes graph matching techniques in order to identify similar subgraphs in the on-line and the off-line graph and based on the latter to suggest future navigation paths and shots. Another approach could be based on incremental on line aggregation of gaze movements of the same user in order to generate gaze-based features of the viewed items for a certain time period. Then, by employing a classifier that discriminates relevant from non-relevant shots (built with aggregated user feedback) we can recommend relevant items. Finally, in a more user-oriented approach, we could train a user-focused classifier in real time using gaze movement and click-through features during a certain time period and consider ground truth any videos viewed (i.e. clicked and watched by the user). After having enough training examples, the classifier could predict and recommend interesting shots for the upcoming retrieval sessions of this user.

#### **7.2.2. Detect user interest based on implicit feedback**

A challenging future work would be to detect user interest by modelling efficiently and combining the heterogeneous implicit feedback including the clicks the gaze movements and further enhance them with additional involuntary user responses such as EEG signals and other biometric data measured using the appropriate sensor.

Through the application of a multi-modal analysis sensory data will be aggregated and used to train user models capable of discriminating between different cognitive behaviours and responses. For all categories of sensory information feature selection will be performed, using as a selection criterion the information gain of each feature and conclude to a representative set of features, which will be used to train a classifier.

The application of information fusion at a decision level will also be part of the methodological approach. Specifically late fusion will be considered (i.e. at decision level) to accept or reject decisions of individual modalities. A majority vote scheme can be employed to make the final judgment.

### 7.2.3. Dynamic multimedia content modelling

Finally, a more broad research activity that could be triggered by the proposed thesis is the development of a dynamic multimodal representation schema of multimedia objects. This could build upon a static content-based representation layer, and will be complemented and enhanced by a dynamic layer of implicit feedback (in terms of past user interaction and affective behaviour) features. To provide an affectively enriched representation of multimedia, including also the information by past user interaction and implicit feedback, a two layer representation could be considered:

- a. The static layer, which will include content-based information. Specifically, this layer would follow a 2-level pyramidal representation schema. The bottom level will include low-level feature descriptors based on visual, motion and audio information, etc. The top layer will include a set of descriptors that provides information for audiovisual events (e.g. dog barking, people walking) or concepts (car, mountain, etc.), in the form of detection scores.
- b. The dynamic layer will be formed by means of automatic annotation based on the identification of user interest from past user interaction and physical behaviour, as expressed by eye movements, EEG signals and biometric sensor measurements. The interaction will be represented as affinity graphs that interconnect multimedia objects between them and with concepts, which are based either on past user queries or on social interaction.

During retrieval, combination of the affinity graphs with feature vector information could be performed either with machine learning techniques or with late fusion of results.

**BIBLIOGRAPHY**

Adcock, J., Cooper, M., Pickens J. "Experiments in interactive video search by addition and subtraction." *Proceedings of the 7th ACM International Conference on Image and Video Retrieval (CIVR'08)*. Niagara Falls, Canada, 2008, 2008. 465-474.

Ajanki, A., Hardoon, D., R., Kaski, S., Paulamaki, K., Shawe-Taylor, J. "Can eyes reveal interest? Implicit queries from gaze patterns." *User Modeling and User-Adapted Interaction archive* 19, no. 4 (2009): 307-339.

Albanese, M., Chianese, A., Moscato, V. Sansone, L. "A formal model for video shot segmentation and its application via animate vision." *Multimedia Tools and Applications* 24, no. 3 (2004): 253-272.

Allen, E., Horvath, S., Kraft, P., Tong, F., Spiteri, E., Riggs, A., Marahrens, Y. "High Concentrations of LINE Sequence Distinguish Monoallelically-Expressed Genes." *Proceedings of the National Academy of Sciences*. 2003. 9940-9945.

Amit, Y., Geman, D. "Shape quantization and recognition with randomized trees." *Neural Computation* 9, no. 7 (1997): 1545-1588.

Antes, J.R. "The time course of picture viewing." *Journal of Experimental Psychology* 103 (1974): 62-70.

Arapakis, I., Konstas, I., Jose, J. M. "Using facial expressions and peripheral physiological signals as implicit indicators of topical relevance." *Proceedings of the 17th ACM international conference on Multimedia (MM '09)*. New York, USA, 2009.

Arapakis, I., Moshfeghi, Y., Joho, H., Ren, R., Hannah, D., Jose, J. M. "Integrating facial expressions into user profiling for the improvement of a multimodal recommender system." *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'09)*. New York, U.S.A, 2009.

Arapakis, I., Y. Moshfeghi, H. Joho, R. Ren, D. Hannah, and Jose J. M. "Enriching User Profiling with Affective Features for the Improvement of a

- Multimodal Recommender System." *Proceedings of the 8th ACM International Conference for Image and Video Retrieval (CIVR'09)*. Santorini, Greece, 2009.
- Arman, F., Depommier, R., Hsu, A., Chiu, M-Y. "Content-based browsing of video sequences." *Proceedings of the 2nd ACM International Conference on Multimedia (MM'94)*. San Francisco, California, USA, 1994. 97-103.
- Azim-Sadjadi, M.R., Salazar, J., Srinivasan, S. "An adaptable image retrieval system with relevance feedback using kernel machines and selective sampling." *IEEE Transactions on Image Processing* 18 (2009): 1045–1059.
- Ballan, L., Bertini, M. and Serra, G. "Video annotation and retrieval using ontologies and rule learning." *IEEE Multimedia* 17, no. 4 (2010): 80–88.
- Bay, H., Ess, A., Tuytelaars, T., Gool, L. V. "Speeded-up robust feature." *Computer Vision and Image Understanding* 110, no. 3 (2008): 346–359.
- Beitzel, S. M., Jensen, E. C., Lewis, D. D., Chowdhury, A., Frieder, O. "Automatic classification of web queries using very large unlabeled query logs." *ACM Transactions on Information Systems* 25, no. 2 (2007).
- Bentley, J.L. "Multidimensional binary search trees used for associative searching." *Communications of the ACM* 18, no. 9 (1975): 509–517.
- Besiris, D. Laskaris, N., Fotopoulou, F., Economou G. "Key frame extraction in video sequences: a vantage points approach." *Proceedings of the IEEE 9th Workshop on Multimedia Signal Processing (MMSP 2007)*. Chania, Crete, Greece, 2007. 434-437.
- Borlund, P. "The IIR evaluation model: A framework for evaluation of interactive information retrieval systems." *Information Research* 8, no. 3 (2003).
- Boser, B. E., I. Guyon, and V. Vapnik. "A training algorithm for optimal margin classifiers." *Proceedings of the 5th Annual Workshop on Computational Learning Theory (COLT'92)*. Pittsburgh, USA: ACM Press, 1992. 144-152.
- Bosch, A., Zisserman, A., Munoz, X., "Image Classification using Random Forests and Ferns". *Proceedings of the IEEE 11th International Conference on Computer Vision (ICCV 2007)*. Rio de Janeiro, Brazil, 2007. 1-8.

- Breiman, L. "Random Forests." *Machine Learning* 45, no. 1 (2001): 5–32.
- Breiman, L., and Cutler, A. "Random Forests Manual v4.0." Technical, UC Berkeley, USA, 2003.
- Breiman, L., Friedman, J. H., Olshen, R. A., Stone, C. J. *Classification and Regression Trees*. New York, USA: Chapman and Hall, 1984.
- Brooks, P., K. Y. Phang, R. Bradley, D. Oard, R. White, and F. Guimbretire. "Measuring the utility of gaze detection for task modeling: A preliminary study." *Proceedings of the International Conference on Intelligent User Interfaces (IUI'06)*. Sydney, Australia, 2006.
- Buhmann, M. D. (2003), *Radial Basis Functions: Theory and Implementations*, Cambridge University Press, 2003.
- Burkard, R., Dell'Amico, M., Martello, S. *Assignment Problems*. SIAM, 2009.
- Buscher, G., A. Dengel, and L. Van Elst. "Eye movements as implicit relevance feedback." *Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI '08)*. Florence, Italy, 2008. 2991-2996.
- Caruana, R, Karampatziakis, N., Yessenalina, A. "An empirical evaluation of supervised learning in high dimensions." *Proceedings of the 25th International Conference on Machine Learning (ICML'08)*. Helsinki, Finland, 2008. 96-103.
- Castagnos, S., Jones, N., Pu, P. "Eye-tracking product recommenders' usage." *Proceedings of the 4th ACM Conference on Recommender Systems*. Barcelona, Spain, 2010. 29–36.
- Chang, C., and C. Lin. 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.
- Chang, S-F., R. Manmatha, and T-S. Chua. "Combining text and audio-visual features in video indexing." *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '05)*. Philadelphia, USA, 2005. 1005-1008.

Claypool, M., P. Le, M. Waseda, and D. Brown. "Implicit Interest Indicators." *Proceedings of the ACM Intelligent User Interfaces Conference (IUI'01)*. Santa Fe, New Mexico, USA, 2001. 14-17.

Cowell, A., Hale, K., Berka, C., Fuchs, S., Baskin, A., Jones, D., Davis, Johnson, R., Patch, R., Marshall, E. "Brainwave- Based Imagery Analysis." *Digital Human Modeling: Trends in Human Algorithms*, 2008: 17–27.

Craswell, N., Szummer, M. "Random walks on the click graph." *Proceedings of the 30th Annual International ACM SIGIR '07*. Amsterdam, The Netherlands, 2007. 239-246.

Dijkstra, E. W. "A note on two problems in connexion with graphs." *Numerische Mathematik* 1, no. 1 (1959): 269-271.

Diriye, A., Zagorac, S., Little, S., Rueger, S. "NewsRoom: An Information-Seeking Support System for News Videos." *Proceedings of the 11th ACM SIGMM International Conference on Multimedia Information Retrieval (MIR'10)*. New York, NY, USA, 2010. 377-380.

Divakaran, A., Radhakrishnan, R., Peker, K.A. "Video summarization using descriptors of motion activity: A motion activity based approach to key-frame extraction from video shots." *Journal of Electronic Imaging* 10, no. 4 (2001): 909-916.

Duchowski, A. T. "A Breadth-First Survey of Eye Tracking Applications." *Behavior Research Methods, Instruments, & Computers (BRMIC)* 34, no. 4 (2002): 455-470.

Ekman, P., Friesen, W. V., O'Sullivan, M., Chan, A., Diacoyanni-Tarlatzis, I., Heider, K., Krause, R., LeCompte, W. A., Pitcairn, T., Ricci-Bitti, P. E. "Universals and cultural differences in the judgments of facial expressions of emotion." *Journal of personality and social psychology* 53, no. 4 (1987): 712-717.

Eom, M., Choe, Y. "Fast Extraction of Edge Histogram in DCT Domain Based on MPEG7." *Proceedings of International Conference on Enformaticka, Systems Sciences and Engineering (ESSE 2005)*. Istanbul, Turkey, 2005. 209-212.

Faro, A, D Giordano, C. Pino, and C. Spampinato. "Visual attention for implicit relevance feedback in a content based image retrieval." *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. Texas, Austin, 2010. 73-76.

Floyd, R.W. "Algorithm 97: Shortest Path." *Communications of the ACM* 5, no. 6 (1962): 345-345.

Huiskes, M. J., Thomee, B. and Lew, M.S. "New trends and ideas in visual concept detection: the MIR flickr retrieval evaluation initiative in MIR." *Proceedings of the International Conference on Multimedia Information Retrieval (MIR'10)*. New York, USA, 2010.

Geisler, G., Marchionini, G., Wildemuth, B.M., Hughes, A., Yang, M., Wilkens, T., Spinks, R. "Video browsing interfaces for the open video project." *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, Extended abstracts on Human factors in computing systems*. New York, USA, 2002. 514–515.

Gerson, A., Parra, L., Sajda, P. "Cortically coupled computer vision for rapid image search." *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 14, no. 2 (2006): 174–179.

Giacinto, G., and F. Roli. "Instance-Based Relevance Feedback for Image Retrieval". Vol. 17, in *Advances in Neural Information Processing Systems*, edited by L. K. Saul, Y. Weiss and L. Bottou, 489-496. MIT Press, 2005.

Gkalelis, N., Mezaris, V. and Kompatsiaris, I. "High-level event detection in video exploiting discriminant concepts." *Proceedings of the 9th International Workshop on Content-based Multimedia Indexing (CBMI '11)*. Madrid, Spain, 2011.

Granka, L.A., Joachims, T. and Gay, G. "Eye-tracking analysis of user behavior in WWW search." *Proceedings of the 27th Annual International ACM SIGIR conference on Research and Development in Information Retrieval (SIGIR '04)*. New York, NY, USA, 2004. 478-479.



Gurrin, C., D. Johansen, and A. F. Smeaton. "Supporting Relevance Feedback in Video Search." *Proceedings of the 28th European Conference on Information Retrieval (ECIR 2006)*. London, UK, 2006. 561-564.

Gutmann, A. "R-trees: a dynamic index structure for spatial searching." *Proceedings of the ACM International Conference on Management and Data (SIGMOD '84)*. New York, USA, 2004. 47-57.

Haggerty, A., White, R. W. Jose, J. M. "NewsFlash: Adaptive TV News Delivery on the Web." *Proceedings of the 1st International Workshop on Adaptive Multimedia Retrieval (AMR '03)*. Hamburg, Germany: Springer, 2004. 72-86.

HajiMirza, S. N. H., Proulx, M., Izquierdo, E. "Gaze movement inference for user adapted image annotation and retrieval." *Proceedings of the 2011 ACM workshop on Social and behavioural networked media access (SBNMA'11)*. Arizona, USA, 2011. 27-32.

Hajimirza, S.N., and E. Izquierdo. "Gaze movement inference for implicit image annotation." *Proceedings of the 11th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'10)*. Desenzano del Garda, Italy, 2010.

Hancock-Beaulieu, Stephen E. Robertson and Micheline. "On the evaluation of IR systems." *Information Processing and Management: an International Journal* 28, no. 4 (1992): 457-466.

Hanjalic A., Zhang, H.J. "An integrated Scheme for Automated Abstraction Based on Unsupervised Cluster-Validity Analysis." *IEEE Transactions On Circuits and Systems for Video Technology* 9, no. 8 (1999): 1280-1289.

Hanjalic, A. "Adaptive Extraction of Highlights From a Sport Video Based on Excitement Modeling." *IEEE Transactions on Multimedia* 7, no. 6 (2005): 1114-1122.

Hanjalic, A. "Shot-boundary detection: unraveled and resolved?" *IEEE Transactions on Circuits and Systems for Video Technology* 12 (2002): 90-105.

Hardoon, D. R., J. Shawe-Taylor, A. Ajanki, K. Puolamaki, and S. Kaski. "Information retrieval by inferring implicit queries from eye movements."

*Proceedings of the 11th International Conference on Artificial Intelligence and Statistics (AISTATS'07)*. San Juan, Puerto Rico, 2007.

Hardoon, D.R., and K. Pasupa. "Image Ranking with Implicit Feedback from Eye Movements." *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. Austin, Texas, 2010. 291-298.

Hauptmann, A., G. "Lessons for the Future from a Decade of Informedia Video Analysis Research." *Proceedings of the 4th Conference on Image and Video Retrieval (CIVR' 2005)*. Berlin, Germany: Springer, 2005. 1-10.

Hess, E. and Polt, J. "Pupil size as related to interest value of visual stimuli." *Science* 132 (1960): 349-350.

Hess, E. and Polt, J. "Pupil size in relation to mental activity during simple problem-solving." *Science* 143 (1964): 1190-1192.

Ho, T. K. "Random Decision Forest." *Proceedings of the 3rd International Conference on Document Analysis and Recognition (ICDAR'95)*. Montreal, QC, 1995. 278–282.

Ho, T. K. "The Random Subspace Method for Constructing Decision Forests." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20, no. 8 (1998): 832–844.

Hopfgartner, F., and J. Jose. "Evaluating the implicit feedback models for adaptive video retrieval." *Proceedings of the International Conference on Multimedia Information Retrieval (MIR'07)*. Augsburg, Bavaria, Germany, 2007. 323-331.

Hopfgartner, F., D. Vallet, M. Halvey, and J. M. Jose. "Search trails using user feedback to improve video search." *Proceedings of the ACM International Conference on Multimedia (MM'08)*. Vancouver, Canada, 2008. 339-348.

Hopfgartner, F., Urruty, T., Hannah, D., Elliott, D., Jose, J.M. "Aspect-based video browsing – a user study." *Proceedings of the IEEE International Conference on Multimedia and Expo (ICME'09)*. New York, USA, 2009. 946–949.

Huang C.-L., Liao, B.-Y. "A robust scene-change detection method for video segmentation." *IEEE Transactions on Circuits and Systems for Video Technology* 11, no. 12 (2001): 1281–1288.

Hubert, L., Arabie, P. "Comparing partitions." *Journal of Classification*, 1985: 193-218.

Hughes, A., T. Wilkens, B. Wildemuth, and G. Marchionini. "Text or Pictures? An Eyetracking Study of How People View Digital Video Surrogates." *Proceedings of the International Conference on Image and Video Retrieval (CIVR 2003)*. Urbana, IL, USA, 2003. 271-280.

Ingwersen, P., Jarvelin, K. *The Turn: Integration of Information Seeking and Retrieval in Context*. Heidelberg, Germany: Springer Verlag, 2005.

Jaccard, P. "Nouvelles recherches sur la distribution florale." *Bulletin de la Société Vaudense des Sciences Naturelles* 44 (1908): 223-270.

Jaimes, A., Pelz, J., B., Grabowski, T., Babcock, J., S. and Chang, S-F. "Using human observer eye movements in automatic image classifiers." *Proceedings of SPIE: Human Vision and Electronic Imaging VI*. San Jose, California, USA, 2001. 373-384.

Jiang, Y., Zeng, X., Ye, G., Bhattacharya, S., Ellis, D., Shah, M. and Chang, S.-F. "Columbia-UCF TRECVID 2010 multimedia event detection: combining multiple modalities, contextual concepts, and temporal matching." *Proceedings of TRECVID 2010 Workshop*. Gaithersburg, MD, USA, 2010.

Joachims, T, L Granka, B Pan, H Hembrooke, and Gay G. "Accurately interpreting clickthrough data as implicit feedback." *Proceedings of the 28th annual international ACM SIGIR'05*. Salvador, Brazil, 2005. 154-161.

Joachims, T. "A Support Vector Method for Multivariate Performance Measures." *Proceedings of the International Conference on Machine Learning (ICML '05)*. Bonn, Germany, 2005.

Joachims, T.. "Optimizing Search Engines Using Clickthrough Data." *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD '02)*. Edmonton, Alberta, Canada, 2002. 133-142.

Joachims, T.. "Training Linear SVMs in Linear Time." *Proceedings of the ACM Conference on Knowledge Discovery and Data Mining (KDD'06)*. Philadelphia, USA, 2006.

Joachims, T. "Unbiased evaluation of retrieval quality using clickthrough data." Technical Report, Department of Computer Science, Cornell University, 2002.

Joho, H., Jose, J. M., Valenti, R., Sebe N. "Exploiting facial expressions for affective video summarisation." *Proceedings of the 8th ACM International Conference on Image and Video Retrieval (CIVR '09)*. New York, NY, USA: ACM, 2009.

Joho, H., Staiano, J., Sebe, N., Jose J. "Looking at the viewer: analysing facial activity to detect personal highlights of multimedia contents." *Multimedia Tools and Applications* 51, no. 2 (2010): 505-523.

Josephson, S., and Holmes, M.E. . "Visual Attention to Repeated Internet Images: Testing the Scanpath Theory on the World Wide Web." *Proceedings of Eye Tracking Research & Applications: Symposium 2002, (ACM SIGCHI)*. 2002. 43-51.

Just, M.A. & Carpenter, P.A. "A theory of reading: From eye fixations to comprehension." *Psychological Review* 87 (1980): 329-354.

Kapoor, A., Shenoy, P., Tan, D. "Combining Brain Computer Interfaces with Vision for Object Categorization." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'08)*. Anchorage, Alaska, USA, 2008.

Kasutani, A., Yamada, E. "The mpeg-7 color layout descriptor: a compact image feature description for high-speed image/video segment retrieval." *Proceedings of the International Conference on Image Processing (ICIP'01)*. Thessaloniki, Greece, 2001. 674-677.

Kelly, D., and J. Teevan. "Implicit Feedback for Inferring User Preference: A Bibliography." *SIGIR Forum* 32, no. 2 (2003): 18-28.

Kherfi, M. L., Brahmi D., and Ziou D. "Combining Visual Features with Semantics for a More Effective Image Retrieval." *Proceedings of the 17th*

*International Conference on Pattern Recognition (ICPR'04)*. Cambridge, UK, 2004. 961-964.

Kierkels, J. J. M., Soleymani, M., Pun, T. "Queries and tags in affect-based multimedia retrieval." *Proceedings of the 2009 IEEE international conference on Multimedia and Expo (ICME'09)*. Piscataway, NJ, USA: IEEE Press, 2009. 1436-1439.

Kirkegaard Moe, K., Jensen, J. M. and Larsen, B. "A qualitative look at eye-tracking for implicit relevance feedback." *Proceedings of the 2nd International Workshop on Context-Based Information Retrieval*. Roskilde, Denmark, 2007. 36-47.

Klami, A., Saunders, C., De Campos, T.E. and kaski, S. "Can relevance of images be inferred from eye movements?" *Proceedings of the 1st ACM international conference on Multimedia information retrieval (MIR '08)*. Vancouver, British Columbia, Canada, 2008. 134-140.

Kleinberg, E. "An Overtraining-Resistant Stochastic Modeling Method for Pattern Recognition." *Annals of Statistics* 24, no. 6 (1997): 2319–2349.

Koelsch, S., Kasper, E., Sammler, D., Schulze, K., Gunter, T., Friederici, A. "Music, language and meaning: brain signatures of semantic processing." *Nature Neuroscience* 7 (2004): 302–307.

Koelstra, S., Muehl, C. Patras, I. "EEG analysis for implicit tagging of video data." *Proceedings of Workshop on Affective Brain-Computer Interfaces (ABCI'09)*. Amsterdam, The Netherlands, 2009. 27-32.

Kozma, L., A. Klami, and S. Kaski. "GaZIR: gaze-based zooming interface for image retrieval." *Proceedings of the 11th International Conference on Multimodal interfaces (ICMI-MLMI '09)*. Cambridge, MA, USA, 2009. 305-312.

Krenz, W, M Robin, S Barez, and L. W. Stark. "Neurology model of the normal and abnormal human pupil." *IEEE Trans. Biomed. Eng* BME-32, no. 10 (1985): 817–825.

Kuhn, H., W. "The Hungarian Method for the assignment problem." *Naval Research Logistics Quarterly* 2 (1955): 83–97.

Lew, M. S., N Sebe, C. Djeraba, and R. Jain. "Content-based Multimedia Information Retrieval: State of the Art and Challenges." *ACM Transactions on Multimedia Computing, Communications, and Applications* 2, no. 1 (2006): 1-19.

Liang, Z., H. Fu, Y Zhang, Z Chi, and D. Feng. "Content-based image retrieval using a combination of visual features and eye tracking data." *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. Austin, Texas, 2010. 41-44.

Lloyd., S. P. "Least squares quantization in PCM." *IEEE Transactions on Information Theory* 28, no. 2 (1982): 129–137.

Loewenfeld, I., and O. Lowenstein. *The Pupil: Anatomy Physiology and Clinical Applications*. Detroit: Wayne State Univ. Press, 1993.

Lowe, D. "Object recognition from local scale-invariant features." *Proceedings of IEEE conference on computer vision and pattern recognition (CVPR'99)*. Los Alamitos, CA, 1999.

Luck, S. J. *An Introduction to the Event-Related Potential Technique*. The MIT Press, 2005.

Manjunath, B. S., Salembier, P., and Sikora T. *Introduction to MPEG-7: Multimedia Content Description Language*. Wiley, 2002.

McCamy, C., Marcus, H., Davidson, J. "A color rendition chart." *Journal of Applied Photographic Engineering* 2, no. 3 (1976): 95–99.

Meila, M. "Comparing clusterings: an axiomatic view." *Proceedings of the 22nd international conference on Machine learning (ICML'05)*. New York, NY, USA: ACM, 2005. 577-584.

Metzler, D., Strohman, T., Turtle, H. and Croft W. B. "Indri at TREC 2004: Terabyte Track." *Proceedings of the Text REtrieval Conference (TREC 2004)*. Gaithersburg, USA, 2004.

Mezaris, V., Dimou, A., Kompatsiaris, I. "On the use of feature tracks for dynamic concept detection in video." *Proceedings of the 2010 IEEE International Conference on Image Processing (ICIP' 10)*. Hong Kong, China, 2010. 4697-4700.

Mezaris, V., et al. "The SCHEMA Reference System: An Extensible Modular System for Content-based Information Retrieval." *Proceedings of the 6th Workshop on Image Analysis For Multimedia Interactive Services (WIAMIS'05)*. Montreux, Switzerland, 2005.

Mikolajczyk, K., Schmid, C. "Performance evaluation of local descriptors." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 27, no. 10 (2005): 1615–1630.

Moraveji, N. "Improving video browsing with an eye-tracking evaluation of feature-based color bars." *Proceedings of the 4th ACM/IEEE-CS joint conference on Digital libraries*. Tuscon, AZ, USA, 2004. 49-50.

Morrison S. Jose, J. "A comparative study of online news retrieval and presentation strategies." *Proceedings of the IEEE 6th International Symposium on Multimedia Software Engineering (ISMSE '04)*. Miami, Florida, Washington, DC, USA, 2004. 403-409.

Moumtzidou, A., Dimou, A., King, P., Vrochidis, S., Angeletou, A., Mezaris, V., Nikolopoulos, S., Kompatsiaris, I., Makris, L. "ITI-CERTH participation to TRECVID 2009 HLF and Search." *Proceedings of TRECVID 2009 Workshop*. Gaithersburg, MD, USA, 2009.

Moumtzidou, A., Dimou, A., Gkalelis, N., Vrochidis, S., Mezaris, V. and Kompatsiaris, I. "ITI-CERTH participation to TRECVID 2010." *Proceedings of TRECVID 2010 Workshop*. Gaithersburg, MD, USA, 2010.

Moumtzidou, A., Sidiropoulos, P., Vrochidis, S., Gkalelis, N., Nikolopoulos, S., Mezaris, V., Kompatsiaris, I., Patras, I. "ITI-CERTH participation to TRECVID 2011." *Proceedings of TRECVID 2011 Workshop*. Gaithersburg, MD, USA, 2011.

*MPEG-7 XM software.*

[http://www.lis.ei.tum.de/research/bv/topics/mmdb/e\\_mpeg7.html](http://www.lis.ei.tum.de/research/bv/topics/mmdb/e_mpeg7.html).

Nam, B., Sussman, A. "A comparative study of spatial indexing techniques for multidimensional scientific datasets." *Proceedings of the 16th International*

*Conference on Scientific and Statistical Database Management*. Santorini, Greece, 2004. 171–180.

Naphade, M., R., Smith, J., R., Tesic, J., Chang, S-F., Hsu, W., H., Kennedy, L., S., Hauptmann, A., G. and Curtis, J. "Large-Scale Concept Ontology for Multimedia." *IEEE MultiMedia* 13, no. 3 (2006): 86-91.

Nichols, D. M. "Implicit ratings and filtering." *Proceedings of the 5th DELOS Workshop on Filtering and Collaborative Filtering*. Hungary, 1998. 31-36.

Nikolopoulos, S., Zafeiriou, S., Nikolaidis, N., Pitas, I. "Image replica detection system utilizing R-trees and linear discriminant analysis." *Pattern Recognition* 43, no. 3 (2010): 636-649.

Over, P., Awad, G., Fiscus, J., Michel, M., Smeaton, A., Kraaij W. "TRECVID 2009 - Goals, Tasks, Data, Evaluation Mechanisms and Metrics." *Proceedings of TRECVID 2009 Workshop*. Gaithersburg, MD, USA, 2009.

Oyekoya, O. K., and F. W. M. Stentiford. "Eye tracking as a new interface for image retrieval." *BT Technology Journal* 22, no. 3 (2004): 161-169.

Oyekoya, O., and F Stentiford. "Exploring Human Eye Behaviour using a Model of Visual Attention." *Proceedings of the 17th International Conference on (ICPR'04)*. Washington, DC, USA, 2004. 945-948.

Oyekoya, O., and F. Stentiford. "Perceptual Image Retrieval Using Eye Movements ." *Advances in Machine Vision, Image Processing, and Pattern Analysis*, 2006: 281-289.

Pantic, M. Vinciarelli, A. "Implicit human-centered tagging." *IEEE Signal Processing Magazine* 26, no. 6 (2009): 173-180.

Pass, G., Zabih, R., Miller, J. "Comparing images using color coherence vectors." *Proceedings of the ACM International Conference on Multimedia (MM'96)*. Boston, Massachusetts, USA, 1996. 65–73.

Pasupa, K., Saunders, C., Szedmak, S., Klami, A., Kaski, S., Gunn, S. "Learning to rank images from eye movements." *Proceedings of IEEE 12th International Conference on Computer Vision (ICCV'2009) Workshop on Human-Computer Interaction (HCI'2009)*. Kyoto, Japan, 2009. 2009-2016.



Patel, N. V., Sethi, I.,K. "Video shot detection and characterization for video databases." *Pattern Recognition* 30, no. 4 (1997): 583-592.

Pavan, M., Pelillo, M. "A new graph-theoretic approach to clustering and segmentation." *Proceedings of IEEE Conference on Computer Vision and Patter Recognition (CVPR 2003)*. Madison, USA, 2003. 762-768.

Pedersen, T., Patwardhan, S., Michelizzi, J. "Wordnet::similarity-Measuring the Relatedness of Concepts." *Proceedings of the 19th National Conference on Artificial Intelligence (AAAI'04)*. California, USA, 2004. 1024-1025.

Ponte, J., Croft W. B. "A language modelling approach to information retrieval." *Proceedings of the 21st International ACM SIGIR Conference on Research and Development in Information Retrieval*. Melbourne, Australia: ACM, 1998. 275-281.

Porter, M.F. "An algorithm for suffix stripping." *Program* 14, no. 3 (1980): 130-137.

Privitera, C., L. Renninger, T. Carney, S. Klein, and M. Aguilar. "Pupil dilation during visual target detection." *Proceedings of the SPIE Annual Symposium on Electronic Imaging: Science and Technology (SPIE'08)*. San Francisco, California, USA, 2008. 68060T-1–68060T-11.

Puolamaki, K., J. Salojarvi, E. Savia, J. Simola, and S. Kaski. "Combining eye movements and collaborative filtering for proactive information retrieval." *Proceedings of the 28th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'05)*. Salvador, Brazil, 2005. 146–153.

Qian M., Aguilar M., Zachery K., Privitera C., Klein S., Carney T., Nolte L. "Decision-level fusion of EEG and pupil features for single-trial visual detection analysis." *IEEE Trans Biomed Eng* 56, no. 7 (2009): 1929-1937.

Radlinski, F, and T. Joachims. "Query chains: learning to rank from implicit feedback." *Proceedings of the 11th ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*. Chicago, Illinois, USA, 2005. 239-248.

Radlinski, F., M. Kurup, and T. Joachims. "How Does Clickthrough Data Reflect Retrieval Quality?" *Proceedings of the ACM Conference on Information and Knowledge Management (CIKM'08)*. California, USA, 2008. 43-52.

Ramanathan, S., Katti, H., Huang, R., Chua, T.-S., Kankanhalli, M. "Ramanathan, S., Katti, H., Huang, R., Chua, T.-S., Kankanhalli, M.: Automated localization of affective objects and actions in images via caption text-cum-eye gaze analysis." *Proceedings of the 17th ACM International Conference on Multimedia (MM'09)*. Beijing, China, 2009. 729-732.

Rayner, K. "Eye movements in reading and information processing." *Psychological Bulletin* 124 (1998): 372–252.

Rui, Y., Huang, T., S., Methotra, S. "Relevance Feedback Techniques in Interactive Content-Based Image Retrieval." *Proceedings of the 6th Conference on Storage and Retrieval for Image and Video Databases (SPIE'98)*. San Jose, California, USA, 1998. 25-36.

Santella, A., Agrawala, M., DeCarlo, D., Salesin, D., Cohen, M. "Gaze-based interaction for semi-automatic photo cropping." *Proceedings of the SIGCHI conference on Human Factors in computing systems (CHI '06)*. Montreal, Canada, 2006. 771-780.

Schoeffmann, K., Hopfgartner, F., Marques, O., Boeszoermenyi, L., Jose, J. "Video browsing interfaces and applications." *SPIE Reviews*, 2010.

Sebe, N., M. S Lew, X. Zhou, T. S. Huang, and Bakker E. M. "The state of the art in image and video retrieval." *2nd International Conference on Image and Video Retrieval (CIVR'03)*. Urbana, Champaign, IL, USA, 2003. 1-8.

Seo, Y-W. and Zhang, B-T. "Learning user's preferences by analyzing web-browsing behaviors." *Proceedings of the 4th International Conference on Autonomous Agents*. Barcelona, Spain: ACM Press, 2000. 381-387.

Shi T, Horvath S. "Unsupervised Learning with Random Forest Predictors." *Journal of Computational and Graphical Statistics* 15, no. 1 (2006): 118-138.

Shinoda, Morita M. and. "Information filtering based on user behaviour analysis and best match text retrieval." *Proceedings of the 17th Annual International*

*ACM-SIGIR Conference on Research and Development in Information Retrieval*. Dublin, Ireland: ACM/Springer, 1994. 272-281.

Smeaton, A., Over, P., Kraaij, W. "Evaluation Campaigns and TRECVID." *Proceedings of the 8th ACM International Workshop on Multimedia Information Retrieval (MIR '06)*. Santa Barbara, California, USA, 2006. 321-330.

Smith, J. D., G. A. Masek, L. Y. Ichinose, T. Watanabe, and L. W. Stark. "Single neuron activity in the pupillary system." *Brain Res* 24 (1970): 219–234.

Snoek, C., G., M., Worring, M., Koelma, D., C., Smeulders, A., W. M. "A Learned-Driven Paradigm for Interactive Video Retrieval." *IEEE Transactions on Multimedia* 9, no. 2 (2007): 280-292.

Snoek, C.G.M., and M. Worring. "Multimodal Video Indexing, a Review of the State-of-the-art." *Multimedia Tools and Applications* 25, no. 1 (2005): 5-35.

Stanley J. C., Campbell D. C. *Experimental and Quasi-Experimental Design for Research*. Monterey, CA: Wadsworth Publishing, 1963.

Strehl, A., Ghosh, J. "Cluster ensembles - a knowledge reuse framework for combining multiple partitions." *Journal of Machine Learning Research* 3 (2002): 583-617.

Strohman, T., Metzler, D., Turtle, H., Croft, W. B. "Indri: A language model-based search engine for complex queries." *Proceedings of the International Conference on Intelligence Analysis (IA'04)*. 2004.

Su, J-H., Huang, W-J., Yu, P., Tseng, V.S. "Efficient relevance feedback for content based image retrieval by minning user navigation patterns." *IEEE Transactions on knowledge and data engineering* 23 (2011): 360– 372.

Sull, S. , Kim, J.R., Kim, Y., Chang, H.S., Lee S.U. "Scalable hierarchical video summary and search." *Proceedings of SPIE Conference on Storage and Retrieval for Media Databases*. San Jose, California, USA, 2001.

Tao, D., Tang, X. Li, X., Wu X. "Asymmetric bagging and random ubspace for support vector machines-based relevance feedback in image retrieval." *IEEE Transactions on pattern analysis and machine intelligence* 28 (2006): 1088–1099.

*The Lemur Project*. <http://www.lemurproject.org/>.

Tong, S., Chang, E. "Support vector machine active learning for image retrieval." *Proceedings of the ACM International Conference on Multimedia (MM'01)*. Ottawa, Ontario, Canada, 2001. 107-118.

Tsamoura, E., Mezaris, V., Kompatsiaris, I. "Gradual transition detection using colour coherence and other criteria in a video shot meta-segmentation framework." *Proceedings of IEEE International Conference on Image Processing, Workshop on Multimedia Information Retrieval (ICIP-MIR '08)*. San Diego, USA, 2008. 45-48.

Tsekeridou, S., Pitas, I. "Content-based video parsing and indexing based on audiovisual interaction." *IEEE Transactions on Circuits and Systems for Video Technologies* 11, no. 4 (2001): 522– 535.

Tsikrika, T., Diou, C., de Vries, A.P., Delopoulos, A. "Image annotation using clickthrough data." *Proceedings of the 8th ACM International Conference on Image and Video Retrieval (CIVR'09)*. Santorini, Greece, 2009.

Turtle, H., Croft, W. B. "Evaluation of an inference network based retrieval model." *Transactions Information Systems* 9, no. 3 (1991): 187-222.

Urban J., Jose, J.M., Van Rijsbergen, C.J. "An Adaptive Technique for Content-Based Image Retrieval," *Multimedia Tools and Applications* 31 (2006):1-28.

Vallet, D., F. Hopfgartner, and J. M. Jose. "Use of Implicit Graph for Recommending Relevant Videos: A Simulated Evaluation." *Proceedings of the Annual European Conference on Information Retrieval (ECIR'08)*. Glasgow, Scotland, 2008. 199-210.

Vapnik., C. Cortes and V. "Support-vector network." *Machine Learning* 20 (1995): 273-297.

Vinh, N.X., Epps, J., Bailey, J. "Information theoretic measures for clusterings comparison: is a correction for chance necessary?" *Proceedings of the 26th Annual International Conference on Machine Learning (ICML'09)*. Montreal, Canada, New York, USA, 2009. 1073-1080.

Viviani, P. "Chapter 8." Chap. 8 in *Eye Movements and Their Role in Visual and Cognitive Processes*, edited by E In Kowler. Amsterdam, The Netherlands: Elsevier Science, 1990.

Vrochidis, S., C. Doulaverakis, A. Gounaris, E. Nidelkou, L. Makris, and I. Kompatsiaris. "A Hybrid Ontology and Visual-based Retrieval Model for Cultural Heritage Multimedia Collections." *International Journal of Metadata, Semantics and Ontologies* 3, no. 3 (2008): 167-182.

Vrochidis, S., Kompatsiaris, I., Patras, I. "Exploiting Implicit User Feedback in Interactive Video Retrieval." *Proceedings of the 11th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS'10)*. Desenzano del Garda, Italy, 2010.

Vrochidis, S., Kompatsiaris, I., Patras, I. "Optimizing Visual Search with Implicit User Feedback in Interactive Video Retrieval." *Proceedings of the ACM International Conference on Image and Video Retrieval (CIVR 2010)*. Xi'an, China, 2010. 274-281.

Vrochidis, S., Kompatsiaris, I., Patras, I. "Utilizing Implicit User Feedback to Improve Interactive Video Retrieval." *Advances in Multimedia* (Hindawi) 2011 (2011): 18 pages.

Vrochidis, S., Moumtzidou, A., King, P., Dimou, A., Mezaris V., Kompatsiaris, I. "VERGE: A video interactive retrieval engine." *Proceedings of the 8th International Workshop on Content-Based Multimedia Indexing (CBMI 2010)*. Grenoble, France, 2010. 142-147.

Vrochidis, S., Patras I., Kompatsiaris, I. "An Eye-tracking-based Approach to Facilitate Interactive Video Search." *Proceedings of 2011 ACM International Conference on Multimedia Retrieval (ICMR2011)*. Trento, Italy, 2011.

Vrochidis, S., Patras, I. Kompatsiaris, I. "Exploiting gaze movements for automatic video annotation." *Proceedings of the 13th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2012)*. Dublin, Ireland, 2012.

Walber, T., Scherp, A., Staab, S. "Identifying Objects in Images from Analyzing the Users' Gaze Movements for Provided Tags." *Proceedings of the 18th international conference on Advances in Multimedia Modeling*. Klagenfurt, Austria, 2012. 138-148.

Wen, J.-R., Nie, J.-Y., Hong-Jiang, Z. "Query clustering using user logs." *ACM Transactions on Information Systems* 20 (2002): 59–81.

White, R., I. Ruthven, and J. M. Jose. "The Use of Implicit Evidence for Relevance Feedback in Web Retrieval." *Proceedings of the 24th BCS-IRSG European Colloquium on IR Research: Advances in Information Retrieval*. Glasgow, UK, 2002. 93-109.

Winder, S., Brown, M. "Learning local image descriptors." *Proceedings of IEEE conference on computer vision and pattern recognition (CVPR'08)*. Minneapolis, Minnesota, USA, 2007. 1–8.

Winder, S., Hua, G., Brown, M. "Picking the best daisy." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'09)*. Miami, Florida, USA, 2009.

Xu, D. and Chang, S.-F. "Recognition Using Kernel Methods with Multilevel Temporal Alignment." *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, no 11 (2008): 1985-1997.

Xu, J. and Croft, B. "Query expansion using local and global document analysis." *Proceedings of the 29th Annual International ACM SIGIR Conference on Research & Development on Information Retrieval (SIGIR'2006)*. Seattle, USA, 1996. 4-11.

Yang, B., T. Mei, X-S. Hua, L. Yang, S-Q. Yang, and M. Li. "Online video recommendation based on multimodal fusion and relevance feedback." *Proceedings of the 6th ACM international conference on Image and video retrieval (CIVR 2007)*. Amsterdam, The Netherlands, 2007. 73-80.

Yarbus, A.L. *Eye Movements and Vision*. New York, USA: Plenum Press, 1967.

- Yazdani, A., Lee, J.-S. Ebrahimi T. "Implicit emotional tagging of multimedia using EEG signals and brain computer interface." *In Proceedings of the 1st ACM SIGMM Workshop on Social media (WSM'09)*. Beijing China, 2009. 81-88.
- Yildizer, E., Metin Balci, A., Hassan, M., Alhadj, R. "Efficient content-based image retrieval using Multiple Support Vector Machines Ensemble." *Journal on Expert Systems with Applications* 39 (2012): 2385-2396.
- Zacks, J., Braver, T., Sheridan, M., et. al. "Human brain activity time-locked to perceptual event boundaries." *Nature Neuroscience* 4, no. 6 (2001): 651-655.
- Zeng, H.-J., He, Q.-C., Chen, Z., Ma, W.-Y., Ma, J. "Learning to cluster web search results." *Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval (SIGIR'04)*. Sheffield, UK, 2004. 210–217.
- Zhang, Q., G Tolias, B. Mansencal, A. Saracoglu, N. Aginako, and et al. "COST292 experimental framework for TRECVID 2008." *Proceedings of TRECVID 2008 Workshop*. Gaithersburg, MD, USA, 2008.
- Zhang, Y., H. Fu, Z. Liang, Z. Chi, and D. Feng. "Eye movement as an interaction mechanism for relevance feedback in a content-based image retrieval system." *Proceedings of the 2010 Symposium on Eye-Tracking Research & Applications*. Austin, Texas, 2010. 37-40.
- Zhou, X. S., Y. Wu, I. Cohen, and Huang T. S. "Relevance Feedback in Content-based Image and Video Retrieval." *Proceedings of the 4th European Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2003)*. London, UK, 2003.
- Zhuang, Y., Rui, Y., Huang, T. S., Metrotra, S. "Adaptive key frame extraction using unsupervised clustering." *Proceedings of IEEE International Conference on Image Processing (ICIP'98)*. Chicago, IL, USA, 1998. 886–890.